

**July, 1982
NEWSLETTER**

\$2⁰⁰

Vol. 2, No. 7

MICHIGAN ATARI COMPUTER ENTHUSIASTS

INSIDE OUR COVER:

- **Midnight Strip**
- **Bats**
- **Baker Street**
- **Screen Graphics to Printer**



DON'T JUST GAZE AT THE STARS . . .



. . . BE ONE!

Turn your programming efforts into real dollars. **soft star** wants to put your original program in Rom Pack. With our experience and expertise you can be assured of all the help you need in achieving the highest R.O.E. (Return On Effort). Our extensive marketing and distribution will give your program maximum national exposure.

So Come Join The Stars.

WRITE:

soft star ltd.

ACQUISITION DEPARTMENT

41715 JOY ROAD

CANTON, MI 48187

JUNE MEETING MINUTES

Sheldon Leemon, Secretary

Since May's minutes weren't published, I would like to mention the visits of our special out of town guests. Earl Rice and Mark Cator, of Atari User Group support, fielded questions from the floor for over an hour. Another honored guest was David Small, who writes the Outpost Atari column for Creative Computing. David was there to show off his new disk drive system for the Atari, which does not use the serial bus, and can therefore read files about 10 times faster than the Atari drives. He will be offering a dual-density model for about \$870, with the special club discount. I did get a chance to see the drive after the meeting, and it works as promised. Thanks to all of our guests.

Also, I would like to mention that in May, awards were given to past officers, and members who had displayed devotion to MACE above and beyond the call of duty. These included past officers Ashby Wolfe, Mark Davids, and Jonathon EArl, software aces Craig Chamberlain, Tom Giese, and Sheldon Leemon, and general do-gooders Chet Gonterman, Don Goldsmith, Todd Mitzner, Thomas Gilfillan, Russel Gill, Richard Schultz, Richard Gizynski, and Gretchan Levitan. MACE really appreciates the support our more active members have given in the past, and is looking forward increasing participation on the part of our new members.

The June meeting was devoted entirely to setting up Special Interest Groups. A lot of enthusiasm was shown, and I'm sure we will all benefit from the chance to get together on a more intimate level than the mass-meeting madness. A full report on the SIGs will be published separately in the newsletter. Much thanks to past-Prez Ashby Woolf for organizing the meeting.

For July, we will be considering some constitutional amendments. As you may know, when the constitution was written about a year ago, MACE had less than 100 members. Now, with about six times that many, we have outgrown some of the original provisions. The changes to be proposed will include:

1) Changing the quorum requirements to a certain percentage of those living within 50 miles of the meeting place to vote, and allowing those requirements to be amended by bylaw if necessary, without a new constitutional amendment. As more out-of-state members join to get our fabulous newsletter, it might become difficult to get a quorum as presently stated.

2) Likewise allowing the spending powers of officers to be amended by bylaw. With the newsletter costing over \$1000 per month, and other expenses increasing with our membership, we should have a flexible way to keep up.

3) Adding new officers to the Board. I propose a SIG Liason to coordinate the Special Interest Groups activity, and splitting Secretary into Corresponding and Recording Secretary posts. The volume of mail has picked up terrifically, and it is hard to keep up.

Those are the major items. All proposed amendments will be read in full at the next meeting, and will be voted on, so plan to attend. ☺

THE MOST IMPORTANT PERIPHERAL YOU'LL EVER OWN

Because the Michigan Atari Computer Enthusiasts provides you with information about your personal computer system. There are special interest groups in Forth, 6502 Assembler, Basic, Games, Education, Business Applications and more. Our monthly meeting offers exciting, informative presentations in the very latest hardware and software products, games and techniques. Our 24 hour Electronic Bulletin Board System becoming a standard for other Atari groups across the nation. A growing diverse local and international membership and award winning proprietary software — all dedicated to providing you with the finest Atari® user support system anywhere.

And don't forget M.A.C.E. Magazine — stimulating, timely, and informative. Quality product reviews, interesting tutorials and 'how to do it' articles offer you page for page one of the most comprehensive Atari® related publications available.

Why not give us a try? Your \$15.00 membership also includes a subscription to M.A.C.E. magazine. Plug us in and boot us up! We think you'll agree that M.A.C.E. is the most important peripheral you'll ever own.

Send check or money order to: M.A.C.E., P.O. Box 2785 Southfield, Michigan 48037.

Call (313) 647-3620

The Center For:

Data Cartridges
Disc Storage/Filing Systems
Rediform Business Forms
Magnetic Media

Typewriter Ribbons
Printer Ribbons
Computer Forms
Acco Binders
Computer Furniture



**MOORE
BUSINESS
CENTER**

Division of Moore Business Forms, Inc.

30600 Telegraph, Suite 1101, Birmingham, MI 48010

Moore Printout Paper is Priced Right! Qualifies for Mix and Match Discount.

Catalog Number	Description	W. & L.	Paper Weight	Parts	Qty. per carton	Price per carton			
						1-2 ctns.	3-5 ctns.	6-9 ctns.	10 ctns. & over
A10025**	line tint	8½ x 11"	13½#	1	3,200	\$27.85	\$25.05	\$23.70	\$22.30
A10116**	line tint	11¾ x 8½"	15#	1	3,200	29.90	26.90	25.40	23.90
A10165**	line tint	14¾ x 8½"	20#	1	2,600	38.50	34.65	32.70	30.80
A10173**	line tint	14¾ x 8½"	18#	1	2,600	31.90	28.70	27.10	25.50
A10181**	line tint	14¾ x 8½"	15#	1	3,200	35.35	31.80	30.05	28.30
A10215**	line tint	14¾ x 11"	20#	1	2,600	38.50	34.65	32.70	30.80
A10223**	line tint	14¾ x 11"	18#	1	2,600	36.85	33.15	31.30	29.50
A10231**	line tint	14¾ x 11"	15#	1	3,200	41.05	36.95	34.90	32.85
A10033**	carbon-interleaved line tint	8½ x 11"	13½#	2	1,400	36.20	32.60	30.80	28.95
A10041**	carbon-interleaved line tint	8½ x 11"	15#	3	850	34.05	30.65	28.95	27.25
A10199**	carbon-interleaved line tint	14¾ x 8½"	13½#	2	1,650	43.20	38.90	36.70	34.55
A10207**	carbon-interleaved line tint	14¾ x 8½"	13½#	3	1,000	41.65	37.50	35.40	33.30
A10249**	carbon-interleaved line tint	14¾ x 11"	13½#	2	1,650	49.90	44.90	42.40	39.90
A10256**	carbon-interleaved line tint	14¾ x 11"	13½#	3	1,000	48.15	43.35	40.90	38.50
A10009*	blank form	8½ x 11"	15#	1	3,200	27.85	25.05	23.70	22.30
A10058*	blank	8½ x 3½"	15#	1	6,400	20.50	18.45	17.40	16.40
A10066* •	blank	9½ x 11"	20#	1	2,600	28.75	25.90	24.45	23.00
A10074* •	blank	9½ x 11"	15#	1	3,200	30.30	27.30	25.75	24.25
A10082* •	carbon-interleaved blank	9½ x 11"	13½#	2	1,650	37.90	34.10	32.20	30.30
A10090* •	carbon-interleaved blank	9½ x 11"	13½#	3	1,000	36.15	32.55	30.70	28.90
A10108* •	carbon-interleaved blank	9½ x 11"	12#	4	800	43.70	39.35	37.15	34.95
A10124* •	blank	12 x 8½"	15#	1	3,200	30.80	27.70	26.20	24.65
A10017**	carbonless blank	8½ x 11"	14#	2	1,800	47.70	42.95	40.55	38.15
A10496**	carbon-interleaved Economy line tint	14¾ x 11"	15#	2	1,350	42.30	38.05	35.95	33.85
A10504**	carbon-interleaved Economy line tint	14¾ x 11"	15#	3	850	42.40	38.15	36.05	33.90
A10512**	carbon-interleaved Economy line tint	14¾ x 11"	12#	4	750	52.70	47.45	44.80	42.15
A10405**	Economy line tint	9½ x 11"	18#	1	2,600	24.85	22.35	21.10	19.90
A10413**	Economy line tint	9¾ x 11"	18#	1	2,600	24.85	22.35	21.10	19.90
A10447**	Economy line tint	14¾ x 8½"	15#	1	3,200	30.80	27.70	26.20	24.65
A10488**	Economy line tint	14¾ x 11"	15#	1	3,200	35.80	32.20	30.45	28.65
A10454**	carbon-interleaved Economy line tint	14¾ x 8½"	15#	2	1,350	36.65	33.00	31.15	29.30
A10462**	carbon-interleaved Economy line tint	14¾ x 8½"	15#	3	850	36.60	32.95	31.10	29.30
A10470**	carbon-interleaved Economy line tint	14¾ x 8½"	12#	4	750	45.65	41.10	38.80	36.50
A10132**	carbonless line tint	14¾ x 8½"	14#	2	1,800	56.80	51.10	48.30	45.45
A10140**	carbonless line tint	14¾ x 8½"	14#	3	1,150	57.65	51.90	49.00	46.10
A10157**	carbonless line tint	14¾ x 8½"	14#	4	850	56.95	51.25	48.40	45.55
A10264**	carbonless line tint	14¾ x 11"	14#	2	1,800	65.65	59.10	55.80	52.50
A10272**	carbonless line tint	14¾ x 11"	14#	3	1,150	66.70	60.05	56.70	53.35
A10280**	carbonless line tint	14¾ x 11"	14#	4	850	65.90	59.30	56.00	52.70
A10298**	carbonless line tint	14¾ x 11"	14#	5	700	73.10	65.80	62.15	58.50
A10306**	carbonless line tint	14¾ x 11"	14#	6	600	78.30	70.50	66.55	62.65
A10314***	carbonless Speediread	14¾ x 8½"	14#	2	1,800	59.15	53.25	50.30	47.30
A10322***	carbonless Speediread	14¾ x 8½"	14#	3	1,150	60.05	54.05	51.05	48.05
A10330***	carbonless Speediread	14¾ x 8½"	14#	4	850	59.25	53.30	50.35	47.40
A10371***	carbon-interleaved Speediread	14¾ x 8½"	15#	2	1,400	44.90	40.40	38.15	35.90
A10389***	carbon-interleaved Speediread	14¾ x 8½"	15#	3	850	43.30	39.00	36.80	34.65
A10397***	carbon-interleaved Speediread	14¾ x 8½"	12#	4	800	53.30	48.00	45.30	42.65
A10348***	Speediread	14¾ x 8½"	20#	1	2,600	34.45	31.00	29.30	27.55
A10355***	Speediread	14¾ x 8½"	18#	1	2,600	32.95	29.65	28.00	26.35
A10363***	Speediread	14¾ x 8½"	15#	1	3,200	36.70	33.05	31.20	29.30
A10439**	Ruled heading Economy	14¾ x 8½"	15#	1	3,200	35.80	32.20	30.45	28.65
A10421**	Economy Speediread	14¾ x 8½"	15#	1	3,200	31.90	28.71	27.10	25.50

*white **white/green ***white/green/brown •strip-off margins

S.I.G. NEWS

MACE has finally organized a number of special interest groups, so that members with common interests can meet more informally than would be possible at our big, crowded general membership meetings. At our June meeting, a number of suggestions for groups were put forth by Ashby Wolfe, and members indicated their choice on a form. We then held the first meeting of these groups, a leader was chosen for each, and a second meeting date and initial goal were chosen.

The groups that registered were:

UTILITIES

Manager: Charles Godfrey
Southfield
559-1272 Home
362-9110 Work

BUSINESS SYSTEMS

Manager: Douglas Perenchio
Warren
776-7626

HARDWARE

Manager: Chris Ratkowski
Detroit
532-5421

EDUCATION

Manager: Mark Davids
St. Clair Shores
774-9709

GAMES

Manager: Stephen Tobias
Sterling Heights
979-5740

BASIC

Manager: Jim Spitzer
Madison Heights
543-0961

FORTH

Manager: Todd Meitzner

Royal Oak
542-1752

COMPUTER SCIENCE

Manager: Gretchan Levitan
Huntington Woods
399-6964

NEW USERS

Manager: Michael Winters
Birmingham
645-2193

GRAPHICS

Manager: Ken Hein
Utica
254-1761

There was also an Assembly-language programming group, but their manager neglected to register with Ashby. We hope that more information about that group will be forthcoming.

As most of the groups will be holding meetings before this issue comes out, we are not printing meeting information here. The most up-to-date source of such information is the group manager.

Of course, Mace is not limiting SIG groups to those formed at the last meeting. Any member who has an idea for a group, and wishes to propose the formation of that group at a meeting is free to do so.

One such group that the Board has suggested is a MACE special interest group. This group would be involved in doing the things necessary to make MACE the biggest and the best group around. Members would take on such key functions as contacting computer stores around the country in order to widen the distribution of the newsletters, putting together public domain software that could be distributed by stores to computer purchasers, thus putting MACE more in the public eye, and organizing large-scale events, like an Atari computer fair. Would you like to see MACE gain national prominence? Would you like to see the biggest Atari computer event in the country staged locally? Then join the MACE SIG! Call Jerry Aamodt, our Vice

continued

President, for more details at 574-1020.

In the future, we hope to have meeting programs put on by the groups, newsletter articles reporting on SIG activites, and a network of "experts" to help members with problems in specific areas. These special interest groups will be one way of making MACE a club where all members have a chance to participate and make a meaningful contribution.

BASIC SIG NEWS

Meeting Schedule:

July 20.....Intermediate

August 12....Beginners

August 24....Intermediate

September 9..Beginners

Time...7:00

For info call Jim Spitzer at 543-0961

Assembler SIG

July Meeting

By Phil Heavin

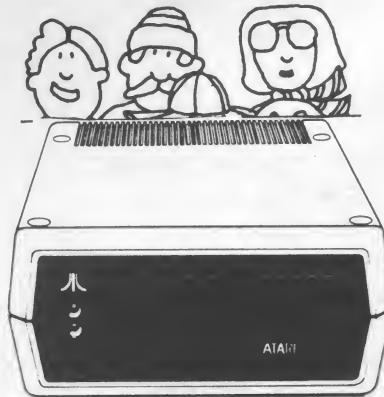
The SIGASM held its first meeting Thursday July 1st at the home of Jim Nixon. This meeting was mostly an organizational type meeting for us to discuss what we hope the SIG can accomplish. Tom Hunt led our meeting and did such a fine job that we volunteered him to be our leader for as long as he can stand it. Since I already had plans to write a few articles for this newsletter about ATARI's new Macro Assembler, I volunteered to be the secretary to report our meetings to MACE. You can at least expect to read a short report each month about our meeting and the announcement of the next meeting. Other times I will be reporting our activities in more detail.

We found that the level of experience in the SIG ranges from people who have been doing assembly programing for over 20 years

to those who are getting their first exposure to assembly language with their ATARI. Because of this we plan to conduct 'class' type sessions at the meetings on two different levels: The beginners level (What's an 'A' register and why do I want to LOAD it?) and an intermediate level for those who understand the mechanics of assembly language but want to learn how to use the beautiful graphics hardware of the ATARI.

In addition to the learning and teaching projects we hope to develop some documentation and programing standards for assembly language programs submitted to the MACE library and develop utility type macros and subroutines for both assembly and BASIC programs. We will probably be working to better define these goals in the next few months so suggestions from all MACE members will be considered. You don't have to want to write assembly programs to be able to suggest valuable programs that this SIG could add to the MACE library.

The next SIGASM meeting will be Thursday August 5th at 7:00 PM at my home in Sterling Heights. Any interested MACE member is invited to attend. You may contact me at 939-6213. M

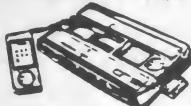


Monitors
PRINTERS
Modems

When It Comes To
ATARI
Getting the best prices
just became as easy

Percom Mini-Disk Drives
for Atari Computers

INTELLIVISION



PAY LESS.

GET
MORE.



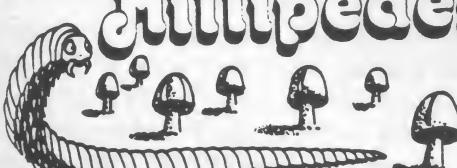
OVER 400 PROGRAMS
IN STOCK

as



Software
ONLY
\$24.95

Millipedes



ATARI 26 K DISC ONLY \$24.95

HIGH PERFORMANCE
FLOPPY DISK **\$1.75**

ATARI 800 with 16K memory \$669
ATARI 400 with 16K memory \$289
ATARI 810 Disk Drive \$449

DISK DRIVE
**HEAD CLEANING
KIT**

\$30. NOW **\$21.95**

EPSON
EPSON AMERICA INC

Hayes

maxell.
DATA PRODUCTS

AMDEK

BAKER STREET BYTES

By Richard Gizynski

The following program is not a new game. It started out as a ten line, GRAPHIC 0 program and grew so that I could show the use of several routines that put commas in numbers, center lines on a screen, concatenate strings (join strings together), use artificial subscripted strings and subscripted variables. A variation of the routines that center strings can be used to turn March's "POOR MAN'S WORD PROCESSOR" into a more useful printer routine.

Now we'll get into the program. Line 30 DIMensions the variables. It also sets a value on the variable YES to 89 and NO to 78. These are the values that are returned by a GET statement for the letters 'Y' and 'N'. To make programs that are user oriented, I often use a GET statement to get information from the keyboard.

Line 40 has a technique shown to us by Bill Wilkinson. It is a quick way of cleaning a large string. In strings DIMensioned to 100 characters or less, you might use:

BLANK\$=" (100 spaces)"

But if you use a really long string, the line 40 method cleans the string at machine speed. This is done by making BLANK\$'s first character a space, opening up the string to its full DIMensioned size and copying the first character into the second character position. The second character, now a blank, is copied into the third character position and so forth.

Line 50 cleans the string NAME\$, which will be used to hold up to five player names.

Line 60 uses a variable 'C' to carry color information to a subroutine at line 1360. This subroutine clears the screen and uses a SETCOLOR 4,C,2 to put the desired color on the screen in full (GRAPHICS 2+16) screen mode. Then the subroutine RETURNS the program to line 70.

Line 70 uses Y to carry the vertical position of the string LINE\$ to a subroutine at line 1120. This subroutine finds out how long string LINE\$ is - LEN(LINE\$) - and subtracts its length from the 20 possible printing spaces available per line in GRAPHICS 2. The result is divided by two and the INTerger (whole number) is found. This gives a value for X which is the horizontal starting position for PRINTing the string. Line 1130 POSITIONs the cursor at the X and Y coordinates and line 1140 PRINTs to device #6 which is the code for the screen. The PRINT statement is abbreviated by using a '?'. Then the program RETURNS to line 80, the statement after our GOSUB on line 70.

Line 80 sends us to another subroutine at line 980 that is used to GET keyboard information. By using subroutines like this, you can use one key to activate your program. You can also protect your program from an 'illegal' entry or an entry that would cause an error message to appear. The first working line of the subroutine POKEs location 694 with 0. This insures that any previous pressing of the Atari (reverse letters) key is cancelled out. POKE 702,64 cancels out the accidental pressing of the caps/lower case key. I often close a Input/Output file before I OPEN it as on line 1000. If you try to OPEN an already open I/O file, you get an error condition but to CLOSE a closed file causes no trouble. Line 1010 GETs the numeric value of the key pressed and 1020 RETURNS.

Lines 90-100 compare the value in KEY to the variables YES and NO. If you pressed a 'Y', line 90 sends you to a series of print statements at lines 1390 to 1500. If you didn't press a 'Y', line 100 checks to see if any key other than a 'N' was pressed to protect against accidental entry.

Lines 120-150 are similar to lines 60-80. They clean the screen and ask a different question. Line 160 is used to translate the key code value of the GET statement to a number that will be carried by the variable PARTICIPANTS. The smaller number is checked by line 170 to be sure it is within the range the game needs. Lines 200 to 220 do the same thing for selection the option that lines 120 to 150 did for PARTICIPANTS.

continued

A FOR-NEXT statement (230-250) is used to zero out the subscripted variable SCORE. When Atari reads the NEXT statement, it checks to see if you have gone through the loop the number of times specified in the FOR statement. If not, you are sent back to the beginning of the loop. Then Atari goes to the following statement, in this case, where GAMES is initialized.

Lines 270 and 280 initialize PLAYER to 1 and check that the current player number isn't greater than the number we want. If we exceed that number, no more names will be asked for and the program GOTOs to the main sequence. Then lines 290 and 300 clear the screen, ask for the current players name and initialize the string that will hold it.

The loop that GETs the players names begins 310. Line 320 sets L to the current LENGTH of string PLAYER\$. Before adding a character to the string, lines 330-350 check to make sure that it isn't a return or delete key. If a return key was pressed, line 330 causes the name held by string PLAYER\$, to be added to the string NAME\$ using the subroutine starting at line 420. A delete key is detected by line 340 or 350 and deletes a letter from the string PLAYER\$ if PLAYER\$ has more than one letter or restarts the routine if it doesn't. Line 360 checks to see that the name doesn't already have the 10 characters that are allowed. Lines 370-380 check the letter to insure its a legitimate character and add it to the string. Line 390 and 400 erase the last PRINT of PLAYER\$ and PRINT the new version. Line 410 send you back to line 310 to get another keystroke. All of lines 320 to 410 were used instead of INPUT PLAYER\$ to protect against entry error.

If the return key was pressed, line 420 adds the string PLAYER\$ to the string NAME\$ by using the numeric variable PLAYER (without the '\$' at the end) to determine what part of string NAME\$ to put string PLAYER\$ in. Numeric PLAYER is multiplied by 10 and then, so we don't waste the first nine possible spaces of string NAME\$, nine is deducted from the result. This gives the starting point in string NAME\$ where string PLAYER\$ will be added. Then 430-440 increments the numeric variable PLAYER and sends the program back to line 280 to see if we have enough names.

Line 460 starts the main loop by adding one to GAMES. Line 470 generates a random number 'RND(0)' and multiplies it by 9999. Numbers after the decimal point are eliminated by the function INTeger. The random number that is generated can be so small that it is still less than one even after being multiplied by 9999 and the INTeger function would return a zero. A one is added to the number to avoid this. Line 480 initiates the HIGH and LOW possible choices and line 490 sets the color and clears the screen.

The numeric variable AMOUNT carries the current value of HIGH to a new subroutine that inserts commas in numbers. At line 1060 AMOUNT is put into the string TEMP\$ so that its length can be measured as in line 1070. Then TEMP\$ put into the string NUMBER\$ that will carry it back to a PRINT routine. The IF statement on line 1080 RETURNS us if TEMP\$ length was only three digits. If there are more than three digits, line 1090 checks for up to six digits and adds a comma in the appropriate place by overwriting the second place from the end of NUMBER\$ with a comma. Then starting at one place from the last of the string NUMBER\$, picks up the part of TEMP\$ that starts two from the end of TEMP\$ length. This is a little confusing to people but very clear to Atari. Remember you are counting backward from the last character not the position after the last character. Line 1100 is only activated if the number has more than 6 digits. The command POP cancels out the need or ability to RETURN from this subroutine. Then LINE\$ carries a message to a subroutine that centers the message on the screen, holds it there long enough to read and then goes back to an early part of the main loop.

Our comma subroutine should return us to line 510 which cleans string LINE\$ then makes it equal to the word "HIGH" and adding our new string NUMBER\$ to it. Then 520 sends us off to a familiar subroutine that centers and prints LINE\$ on the screen. Lines 530-550 do the same thing for the variable LOW.

Lines 560 cleans the string PLAYER\$ and picks the current players name from the string NAME\$. Line 570 lets 10 character names bypass the next routine on lines 580-610 which find the end of the players name by

continued

using a FOR-NEXT loop to check for a space after the name and shorten the name accordingly. Remember, 10 characters including blanks would have been picked up for the first four possible names. Lines 620-650 center and print the name and the word 'GUESS'.

Line 660 sends us off to another subroutine at lines 1160-1340 that works on numbers like the routine at lines 300-410 work for players names. After the variable GUESS is printed to the screen, lines 680-690 check GUESS to make sure it was closer to the correct number, print a message if it wasn't and give the current player another chance. If the guess was in the correct range, lines 700-710 adjust the HIGH-LOW limits of choices and line 720 checks to see if the correct answer was given. If there was a correct answer, the program goes to an ending routine. Otherwise the PLAYER number is incremented at line 730 and line 740 sends the program back for another guess.

Lines 760-970 hold the end routine. 770 decodes a win under option one, adds 1 to the current player's score and bypasses the option two results. If option two was chosen, the correct number gets a "SORRY" message and line 790-800 add one to all of the players scores and deduct one for the booby prize from the currant player. Line 810 leaves a short delay on the screen so either message can be read and 820 clears the screen.

Lines 840-900 use another FOR-NEXT routine to pick each player's name from the string NAME\$ and put it and the player's score in string LINE\$. Then the print position is incremented and LINE\$ is printed by the subroutine starting at 1120. Line 910 holds the scores on the screen long enough for the players to read them and line 920 decides if one number has been presented for each player. If not 930 increments the player and line 940 recycles the guess routine. If the game is finished lines 950-970 ask if you want to play again and act on your answer. ☺

Fred,
 I was over at Binary today.
 They are having a Summer Sale.
all the Atari Software they have is
 marked down at least 25%!
 I picked up a PAC-MAN for only \$33.72 and
 that program you wanted to get from that Mail-Order
 company is less now at Binary.
 They have a great assortment of disks, cassettes, games, and
 educational software.
 Now is the time to get some of those new and old programs
 we wanted.
 I'll meet you there later.
 They're located at 3237 Woodward Avenue, in Berkley
 just South of 12 mile near the Denny's Restaurant.
 P.S. Bring Cash - No Credit Cards are accepted
 Me

BINARY

(313) 548-0533

The Computer Express

FREE Catalog

ATARI 400/800

BUSINESS / PERSONAL

	TCE	LIST
BUDGETMASTER & CHECK BAL	26.99	34.95
BUDGETMASTER & CHECK BAL	30.99	39.95
DATA PERFECT	79.99	99.95
DOW JONES INV. EVAL	95.99	129.95
FILEMANAGER 800	75.99	99.95
LETTER PERFECT	109.99	150.00
MAIL MERGE	22.99	29.00
LETTER PERFECT + MM	205.99	249.95
PERSONAL FINANCE MANG.	58.99	74.95
VISICALC	189.99	250.00



EDUCATION

	TCE	LIST
THE BIG MATH ATTACK	14.99	20.00
COMPU-MATH: FRACTIONS	30.99	39.95
COMPU-MATH: DECIMALS	30.99	39.95
COMPU-READ	30.99	39.95
COMPU-MATH: FRACTIONS	22.99	29.95
COMPU-MATH: DECIMALS	22.99	29.95
COMPU-READ	22.99	29.95
FISHING FOR HOMONYMS	10.99	15.00
GUESSWORD	10.99	15.95
INVITATION TO PROG. #1	19.99	24.95
INVITATION TO PROG. #2	22.99	29.95
INVITATION TO PROG. #3	22.99	29.95
MATH FACTS-LEVEL I	10.99	15.00
MATH FACTS-LEVEL II	10.99	15.00
MATH FACTS-LEVEL III	10.99	15.00
PRESCHOOL FUN	12.99	17.50
SPATIAL RELATIONS	12.99	17.50
SPEED READ PLUS	45.99	59.95
WANTED	10.99	15.00
WORDRACE	19.99	24.95
WORD SCRABBLE	10.99	15.00
WORDMATE	10.99	15.00

T=Tape K=Cartridge E=Disk or Tape
Unmarked Programs are on Disk

ELEPHANT DISKS

	TCE	LIST
SS-SD	\$19.99	
DS-DD	\$29.99	
a box of 10		

ELEPHANT T-SHIRT

	TCE	LIST
S,M,L,XL	\$4.99	

MAXELL DISKS

	TCE	LIST
MD1	\$33.99	
MD2-D	\$52.99	

HEAD CLEANING KIT

	TCE	LIST
\$21.99	List	29.95

DiscSavers

	TCE	LIST
\$5.49	doz.	

BMC 13" COLOR MONITOR w/Audio

	TCE	LIST
\$269	List	29.99

BMC 12" GREEN SCREEN MONITOR

	TCE	LIST
\$89	List	\$99

Hayse SmartModem

	TCE	LIST
\$219	List	27.99

800 Audio Cable

	TCE	LIST
Connects an 800		to a Hi-Fi 7.99

Accessories Too!

ATARI 400/800

TOOLS / ENHANCEMENTS

	TCE	LIST
ASSEMBLER 6502	(T)	19.99
ATARI WORLD		45.99
BASIC A+		59.99
THE BASIC COMPILER		75.99
DATAS 65 2.0		69.99
DISASSEMBLER 6502	(T)	8.99
DISASSEMBLER 6502		10.99
DISKEY		37.99
DISKMANAGER		21.99
EDIT 6502	(K)	165.99
FORTH (QS)		59.99
LISP INTERPETER		109.99
MACRO ASSEMBLER & EDITOR		69.99
MICROSOFT BASIC		69.99
OS/A+		59.99
PILOT	(K)	64.99
SOFTER		37.99



SIGNALMAN MK II MODEM
-Direct Connect
-300 bps F.DPLX
-Cable included
-Talk/Data Swt.
\$87.99 List \$99.95



C.I.TOH
STARWRITER F10
-LETTER QUALITY
-40 cps
-Diablo Daisy
Wheel & Ribbon
Compatible
-15" Carriage
\$1499 * List \$1995



FLIP 'N' FILE
DISK CASE
-Holds 50-60
-Hinged Lid
-Index Dividers
\$21.99 List 29.95

ACCESSORIES

	TCE	LIST
TECHNICAL USER NOTES		22.99
ADVENTURES 1-12 HINT BOOK	6.49	7.95
JOYSTICKS (pair)		15.99
PADDLES		15.99
LE STICK (SALE)		25.99
GRAPHICS TABLET		238.99

C.I.TOH PRO/WRITER

	TCE	LIST
- 120 cps Bi-Dir, Logic Seeking		
- Friction & Tractor Standard		
- N X 9 Matrix		
- 8 Character sizes		
- Proportional spacing		
- Bit & Block Graphics		
- Greek Symbols		
- Mix Text & Graphics		
ASK FOR FACT SHEET/PRINT SAMPLE		

* PARALLEL \$499 List \$795

* SERIAL \$639 List \$895

JULY 82 SUPER SPECIALS

DISK DETECTIVE \$19.99 LIST 29.95

CYPHERBOWL \$33.99 (T) LIST 49.95

THRESHOLD \$25.99 LIST 39.95

SUMMER MADNESS

THE NEXT STEP \$25.99 LIST 39.95

3-D SUPERGRAPHICS \$25.99 (E) LIST 39.95

WORDRACE \$16.99 LIST 24.95

RICOCHET \$12.99 LIST 19.99

ANY GAME BELOW FOR \$19.99 LIST 29.95

PACIFIC COAST HIGHWAY (E)

CRUSH CRUMBLE CHOMP (E)

RASTER BLASTER (D)

CANYON CLIMBER (D)

APPLE PANIC (D)

POOL 1-5 (D)

Specials good until 8/15/82

The Computer Express

M-F 10 AM - 9 PM P.O. Box 569, Dept 1
Sat 10 AM - 6 PM Troy, MI 48099
313 528-1554 FREE CATALOG



Add \$2.00 for shipping. MI residents add 4% tax. Money orders, checks (allow 10 days) accepted. MC/VISA accepted with card # & expir. date. Add \$1.50 for C.O.D. Prices subject to change.

```

10 REM *** NUMBER CHOICE ***
20 REM *** BY RICHARD GIZYNSKI ***
30 DIM PLAYER$(10),NAME$(50),SCORE(5),
NUMBER$(10),BLANK$(50),TEMP$(10),LINE$,
(20),GUESS$(10):YES=89:NO=78:DELAY=200
40 BLANK$ = " " : BLANK$(50) = "
":BLANK$(2,50)=BLANK$
50 NAME$=BLANK$
60 C=4:GOSUB 1360:Y=3:LINE$="DO YOU
WANT":GOSUB 1120
70 Y=4:LINE$="INSTRUCTIONS[Y/N]":GOSUB
1120
80 GOSUB 980
90 IF KEY=YES THEN GOTO 1390
100 IF KEY<>NO THEN GOTO 60
110 REM **** HOW MANY PLAYERS ****
120 C=9:GOSUB 1360:POSITION 3,4
130 Y = 4 : LINE$ = " HOW MANY
PLAYERS":GOSUB 1120
140 Y=5:LINE$="ARE THERE [1-5]":GOSUB
1120
150 GOSUB 980
160 PARTICIPANTS=KEY-48:? CHR$(KEY)
170 IF PARTICIPANTS<1 OR
PARTICIPANTS>5 THEN 120
180 GOSUB 1360
190 REM *** WHICH OPTION ***
200 POSITION 3,4:? #6;"OPTION 1 OR 2"
210 GOSUB 980:OPTION=KEY-48
220 IF OPTION<1 OR OPTION>2 THEN 180
230 FOR I=1 TO PARTICIPANTS
240 SCORE(I)=0
250 NEXT I:GAMES=0
260 REM *** GET PLAYER'S NAMES ***
270 PLAYER=1
280 IF PLAYER=PARTICIPANTS+1 THEN
PLAYER=1:GOTO 450
290 C=9:GOSUB 1360:POSITION 2,4:? #6;"WHAT IS PLAYER ";PLAYER;"'S NAME: ";
300 PLAYER$=BLANK$:PLAYER$=""
310 GOSUB 980
320 L=LEN(PLAYER$)
330 IF KEY=155 THEN 420
340 IF KEY=126 AND L<2 THEN POSITION
8,5:? #6;"":GOTO 300
350 IF KEY = 126 THEN POSITION
8,5:? #6;"":GOTO 390
360 IF L=10 THEN ? #6;" TOO MANY
LETTERS!":DELAY=200:GOSUB 1040:GOTO 280
370 IF KEY<65 OR KEY>90 THEN GOTO 310
380 PLAYER$(L+1)=CHR$(KEY)
390 POSITION 8,5:? #6;BLANK$
400 POSITION 8,5:? #6;PLAYER$
410 GOTO 310
420 NAME$(PLAYER*10-9)=PLAYER$
430 PLAYER=PLAYER+1
440 GOTO 280
450 REM *** MAIN LOOP ***
460 GAMES=GAMES+1
470 NUMBER=INT(RND(0)*9999)+1
480 HIGH=10000:LOW=0
490 C=4:GOSUB 1360
500 AMOUNT=HIGH:GOSUB 1050
510 LINE$=BLANK$ : LINE$ = " HIGH
":LINE$(6)=NUMBER$
520 Y=1:GOSUB 1120
530 AMOUNT=LOW:GOSUB 1050
540 LINE$=BLANK$ : LINE$ = " LOW
":LINE$(5)=NUMBER$
550 Y=10:GOSUB 1120
560 PLAYER$=BLANK$ : PLAYER$=
NAME$(PLAYER*10-9)
570 IF LEN(PLAYER$)=10 THEN GOTO 620
580 PLAYER$(10)=" ";L=0:FOR I=1 TO 10
590 IF PLAYER$(I,I)=" " THEN L=I-1:POP
:I GOTO 610
600 NEXT I
610 PLAYER$=PLAYER$(1,L)
620 LINE$=BLANK$ : LINE$=PLAYER$
630 LINE$(LEN(LINE$)+1)=""S"
640 Y=4:GOSUB 1120
650 POSITION 7,5:? #6;"GUESS"
660 GOSUB 1160:REM GET NUMBER ROUTINE
670 REM *** CHECK ANSWER ***
680 IF AMOUNT>HIGH THEN LINE$="TOO
HIGH":GOTO 1370
690 IF AMOUNT<=LOW THEN LINE$="TOO
LOW":GOTO 1370
700 IF AMOUNT<NUMBER THEN
LOW=AMOUNT
710 IF AMOUNT>NUMBER THEN
HIGH=AMOUNT
720 IF AMOUNT=NUMBER THEN GOTO 760
730 PLAYER=PLAYER+1 : IF
PLAYER>PARTICIPANTS THEN PLAYER=1
740 GOTO 490
750 REM *** ENDING ROUTINE ***
760 LINE$=BLANK$
770 IF OPTION=1 THEN LINE$="YOU WIN":
SCORE(PLAYER)=SCORE(PLAYER)+1:Y=7:
GOSUB 1120:GOTO 810
780 LINE$="SORRY":LINE$(7)=PLAYER$:
Y=7:GOSUB 1120:LINE$="THAT'S THE
NUMBER":Y=8:GOSUB 1120
790 FOR I=1 TO PARTICIPANTS:SCORE(I)=
SCORE(I)+1:NEXT I
800 SCORE(PLAYER)=SCORE(PLAYER)-1
810 DELAY=200:GOSUB 1040
820 GOSUB 1360

```

```

830 REM *** PRINT THE SCORES ***
840 FOR I=1 TO PARTICIPANTS
850 LINE$=BLANK$
860 PLAYER$=NAME$(I*10-9)
870 LINE$=PLAYER$
880 LINE$(12)=STR$(SCORE(I))
890 Y=I+2:GOSUB 1120
900 NEXT I
910 DELAY=400:GOSUB 1040
920 IF GAMES=PARTICIPANTS THEN 950
930 PLAYER=PLAYER+1:IF PLAYER>
PARTICIPANTS THEN PLAYER=1
940 GOTO 460
950 Y=Y+2:LINE$="PLAY AGAIN"
[Y/N]":GOSUB 1120
960 GOSUB 980:IF KEY=YES THEN RUN
970 END
980 REM *** GETTING KEYBOARD INPUT
990 POKE 694,0:POKE 702,64
1000 CLOSE #2:OPEN #2,4,0,"K:"
1010 GET #2,KEY
1020 RETURN
1030 REM **** TIMING SUBROUTINE ***
1040 FOR I=1 TO DELAY:NEXT I:RETURN
1050 REM **** COMMA SUB ****
1060 TEMP$=STR$(AMOUNT)
1070 L=LEN(TEMP$)
1080 NUMBER$=TEMP$:IF L<4 THEN RETURN
1090 IF L<7 THEN NUMBER$(L-2)=",":"
NUMBER$(L-1)=TEMP$(L-2):RETURN
1100 POP :LINE$="OOPS - TOO MANY
DIGITS:GOTO 1370
1110 REM *** CENTER SUBROUTINE ***
1120 X=INT((20-LEN(LINE$))/2)
1130 POSITION X,Y
1140 ? #6;LINE$
1150 RETURN
1160 REM *** INPUT AND PRINT NUMBER
1170 GUESS$=BLANK$
1180 GUESS$=""
1190 GOSUB 980
1200 REM ***CHECK FOR DELETE-RETURN
1210 IF KEY=126 AND LEN(GUESS$)<2 THEN
GUESS$="";NUMBER$=BLANK$:GOTO 1300
1220 IF KEY=126 THEN GUESS$=GUESS$(
1,LEN(GUESS$)-1):GOTO 1270
1230 IF KEY=155 THEN RETURN
1240 KEY=KEY-48
1250 IF KEY<0 OR KEY>9 THEN 1190
1260 GUESS$(LEN(GUESS$)+1)=STR$(KEY)
1270 AMOUNT=VAL(GUESS$)
1280 GOSUB 1060
1290 POSITION 0,6
1300 ? #6;BLANK$
1310 X=INT((20-LEN(NUMBER$))/2)

```

```

1320 POSITION X,6
1330 ? #6;NUMBER$
1340 GOTO 1190
1350 REM *** SET GRAPHICS & COLOR
1360 GRAPHICS 2+16:SETCOLOR
4,C,2:RETURN
1370 Y=7:GOSUB 1120:DELAY=200:GOSUB
1040:POSITION 0,6:?#6;BLANK$;BLANK$:
GOTO 490
1380 REM *** INSTRUCTIONS ***
1390 GRAPHICS 0
1400 POSITION 12,0
1410 ? "INSTRUCTIONS":? ?: "ENTER THE
NUMBER OF PLAYERS AND PLAYER";
1420 ? "NAMES WHEN REQUESTED. ATARI
WILL PICK":? "A NUMBER AT RANDOM FROM
1 TO 9,999"
1430 ? ?: "OPTION ONE -- BE THE FIRST TO
GUESS":? "THE NUMBER ATARI HAS
SELECTED"
1440 ? ?: "OPTION TWO -- MAKE SOMEONE
ELSE":? "GUESS THE NUMBER":?
1450 ? "ATARI WILL TELL YOU IF YOU ARE
TOO":? "HIGH OR TOO LOW. IN BOTH
OPTIONS YOU"
1460 ? "MUST GUESS CLOSER TO THE
CORRECT":? "NUMBER. AFTER THE CORRECT
NUMBER IS"
1470 ? "GUESSED, ANOTHER NUMBER IS
PRESENTED":? "THIS CONTINUES TILL A
NUMBER HAS BEEN"
1480 ? "OFFERED FOR EACH PLAYER":? ?:?
"ATARI WILL ADD COMMAS TO YOUR
NUMBERS"
1490 ? "SO PLEASE DON'T TYPE THEM IN":? ?
"? "PRESS ANY KEY TO CONTINUE";
1500 GOSUB 980:GOTO 120 ⑤

```

M.A.C.E.

The original articles that appear in M.A.C.E. may be reprinted without permission by school and college publications, personal computing club newsletters and nonprofit publications. Each article reprinted must carry a notice to the effect that it is a reprint from M.A.C.E. and give credit to the author. © 1982

Michigan Atari Computer Enthusiasts is a club and users group and is not affiliated with the Atari Company, a Division of Warner Communications Inc.

THE GLASS-TUBE "COLORING BOOK"

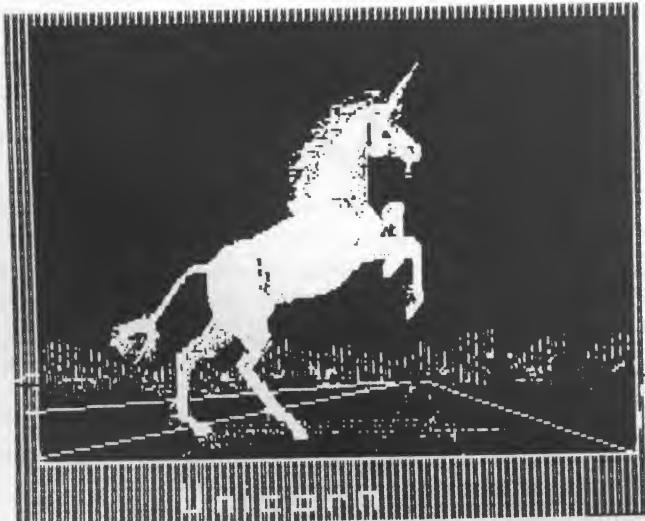
by Linda Peckham

A couple of weeks ago, I bought a couple of new electronic toys. No, I didn't buy a hand-held game. I didn't buy a computer game, either. In fact, the toys aren't games at all. What did I buy? Well, just keep reading.

You know, toys are wonderful things, even when some of them cost quite a bit of money. Especially when they do more than just play games. I'm not much on playing games. (A ten-year-old kid with hyper-reflexes I'm not!) But a toy that will cater to my hobbies, now ...

I started my electronic toy collection a bit more than a year ago. (If you don't know what "toy" I'm talking about, why are you reading this newsletter?) I didn't buy everything at once, of course. (I don't make that much money!) I started with the central "brains" of my collection, and gradually added to it. By the end of the year, I had a reasonably hefty collection, and one of my hobbies was zipping merrily along.

But the other one ... well, it was still in the primitive stage, despite the capability of my toys. What good is a coloring book, if you have to write down exactly how each line goes? So, I bought some instructions to tell the "brain" how to turn the rudder-controls of a starship into an electronic pencil. I bought "Graphics Composer."



My doodler's heart was delighted. Triangles, rectangles, circles, images and fancy patterns, all blazed into colorful glory on my glass tube. No more eraser-smeared paper, no starting over when one small item was out of place. Change colors, even reverse the entire picture, with a couple of keystrokes. Save and redo the picture, over and over again, and it still looks as clean as when I began.

But, there were a few problems. Sure, I could retrieve my pictures when ever I wanted to. But how could I show my friends the results of my work? Lug all my toys around in a couple of suitcases??? I may like my pictures, but that good they're not! What I needed was a way to transfer those images from the glass-tube to the old standby -- paper.

The first step, of course, is to teach the paper-eating toy some new tricks. I persuaded it to munch on a trio of black rectangles with silver legs. That was better, but it was back to the "write - down - how - to - draw - a - line" syndrome. I needed another set of instructions to translate the patterns on the tube to patterns on paper. Enter Datasoft, and "Color Print."

COLOR?! Yep, color! It is a bit awkward; after all, the paper-muncher only thinks in one color -- black. What Color Print does is take the picture, and figure out what color gets printed where. What you do is take one of the carbons (included in the package), and slip it in between two sheets of folded paper. The tricky part is making sure the paper goes back into the muncher in the exact same place everytime. When you're ready, tell Color Print, and it will send the muncher screeching and buzzing down the page. Repeat the process. And, voila! Colored pictures, without multi-thousand-dollar paper munchers! Tired of messing with color? Tell Color Print that, and it will cheerfully ask the muncher to bang out all the colors at once, in glorious black - and - white.

So, with Color Print, I've come full circle. I started off with paper and pencil, and I'm back to paper. My "glass-tube coloring-book" kit isn't as agile as it could be. The "Graphics Composer" doesn't have three-dimensional (drawings) capabilities. And that rudder-controller I mentioned? Eight axes of motion just isn't enough.

continued

I'll probably be adding to my toy kit. Datasoft's been advertising three drawing packages. If I wanted to splurge, I'd get the light-pen, or the digitizer-pad ... I wonder how a trackball would work with "Graphics Composer" ? ... aren't toys wonderful?

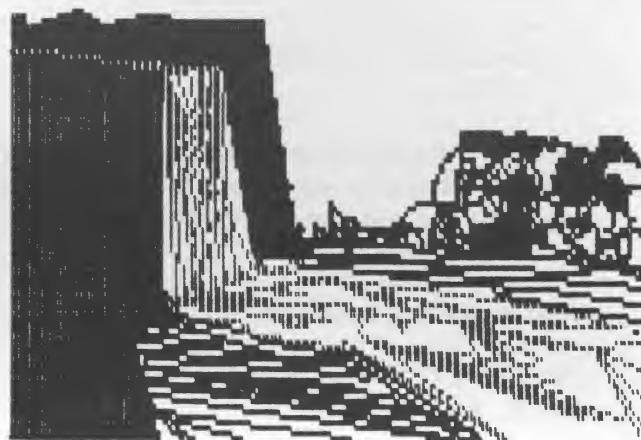
NOTES:

GRAPHICS COMPOSER is produced by Versa Computing, Inc. It comes on disk or cassette, and is composed of five main programs. It draws in Graphics 7 or 8, with joysticks or paddles.

The files produced by the above are not compatible with Color Print. However, it is easy to load the picture with the Basic subroutine provided by the Composer, and then load the Color Print main program.

COLOR PRINT comes on (protected) diskette only, and works with the Epson MX-80 and NEC-8023 printers. Carbons are provided, and refills are available.

The "paper-muncher" is the MX-80, with the new Graphtrax-Plus ROM set. The Plus adds new fonts and (according to Datasoft) faster bit-graphics (compared to Graphtrax). The documentation is a spiral-bound notebook which includes the original documentation, plus new material. It also includes a special section on the Atari software differences. M



CHAOTIC RUMBLINGS

Hello,

You are now hearing from that lucky weight upon whom the onerous and timely burden of communicating with the MACE newsletter has fallen. All seriousness aside, I am the president of the Lansing Area Computer Enthusiasts, or something to acronym LANCE, but LANCE already has a DBA in Michigan, and so does SMAUG, so we are now CHAOS, or something similar if we can get the DBA from the Secretary of State, so we are Capital Hill Atari Owners Society, so CHAOS seems somehow appropriate.

If you push something hard enough it can really move (or fall over). Our local club is about six months old, and by the time we had received your May issue, discussing the confab over opening local chapters, we were one, thanks to some fast negotiating on the parts of Arlan Levitan and our own Ike Hudson. The vote for chapterhood-izationness was unanimous.

So, who are we anyhow? We are CHAOS, and we meet the second Saturday of each month in room C-100 Wells Hall, MSU. We're new, and our average attendance at meetings runs about 25 people. The way we've been growing, we may have to move to a larger room next Fall. Meetings start around 9:00 AM for general B.S. and impromptu program demonstration, with the actual business meeting happening sometime around 10:30. Presentations/demonstrations of a more general nature occur at 10:00 and 11:00. At our August 14 meeting, Brian Goluska will demonstrate printing graphics from the screen, including GRAPHICS 8.

Assuming that I am not impeached for the lighthearted way I handle our communications (or there is a coup), you will be hearing more from us on a monthly basis. Several of us are planning to regularly attend the MACE meetings, so we may soon get to know each other better than as little dots on paper.

Gregory W. Barr
President of CHAOS
PO Box 1343
East Lansing, Mich. 48823
(517)337-8320

ATARI SEQUENTIAL MEMORY FILLER

(My First
Assembly Language Program)

By M. L. CLAYTON

ATARI Basic provides the user a fill function for graphics map mode but there is no equivalent function for machine memory. Described here is a machine language routine to be used in ATARI Basic which sequentially fills consecutive ram location with an integer from 0 to 255.

Because the routine is relocatable and only 56 bytes in length; I usually place the code into a string and use the address of the string as input to the USR call.

An example call would be:

100 FILL = USR(ADR(FILLSTR\$), LOW,
HIGH, VAL)

WHERE:

LOW=256+(PEE(106)-4) -- Points to memory location 1K below ramtop.

HIGH=LOW+1023 -- Points to last location before ramtop.

VAL=0 -- Value to be inserted into specified locations

FILLSTR\$="***the 56 characters appear here ***"

The call would instantly clear the ram reserved for double line resolution player-missile graphics.

A statement of the format

100 FILL=USR(ADR("*****"),
LOW,HIGH,VAL)

with the string imbeded in the statement will also work. If the previous statement is modified to be a subroutine i.e.,

100 FILL=USR(ADR("*****"),
LOW,HIGH,VAL):RETURN

then a statement in the following format will suffice.

The only alternatives to this routine in ATARI Basic are pokes in a "FOR-NEXT" loop (which is slow); and string assignments (which is memory inefficient and requires variable and array pointers to be altered).

For those who keep track of such things; this routine requires the use of page zero memory locations \$D0 and \$d1 (208, 209), and the last two bytes of page six, \$6FE (1790, 1791).

The coded string can be generated by typing the ATASCII character representation of the 56 numerical data values below.

Also the routine can be poked into any 56 free memory locations such as in page six of memory:

```
10 PAGE6 = 1536: FOR I = PAGE 6 TO PAGE6
+ 55 : READ N : POKE I,N : NEXT I
20 DATA 104, 104, 133, 209, 104, 133, 208,
104, 141, 255, 6, 104, 141, 254, 6
30 DATA 104, 104, 168, 162, 0, 152, 129, 208,
56, 173, 254, 6, 229, 208, 208
40 DATA 8, 173, 255, 6, 229, 209, 208, 1, 96,
24, 169, 1, 101, 208, 133, 208
50 DATA 169, 0, 101, 209, 133, 209, 169, 0,
240, 220
```

Then the call would be:

100 FILL=USR(1536,LOW,HIGH,VAL)



DISCOUNT SOFTWARE ACCESSORIES

Check Programs Desired • Many More Available at Discount Prices

	YOUR LIST PRICE		YOUR LIST PRICE
ADVENTURE INTERNATIONAL		AVALON HILL GAME COMPANY	
<input type="checkbox"/> Star Trek 3.5 (d)	24.95	<input type="checkbox"/> Conflict 2500 (t)	16.00
<input type="checkbox"/> Lunar Lander (d)	19.95	<input type="checkbox"/> Planet Miners (t)	16.00
<input type="checkbox"/> Rear Guard (d)	24.95	<input type="checkbox"/> Lords of Karma (t)	20.00
<input type="checkbox"/> Adventure #1 - Adventureland (t)	19.95	<input type="checkbox"/> Empire of the Overmind (t)	30.00
<input type="checkbox"/> Adventure #2 - Pirate's Adv (t)	19.95	<input type="checkbox"/> Tanktics (t)	24.00
<input type="checkbox"/> Adventure #3 - Mission Imposs (t)	19.95	<input type="checkbox"/> Computer Stocks & Bonds (t)	16.00
<input type="checkbox"/> Adventure #4 - Voodoo Castle (t)	19.95		12.00
<input type="checkbox"/> Angle Worms/Crolon Diversion (t)	14.95		
<input type="checkbox"/> Deflection (t)	14.95		
<input type="checkbox"/> Mountain Shoot (t)	14.95		
<input type="checkbox"/> Treasure Quest (t)	19.95		
<input type="checkbox"/> Rear Guard (t)	19.95		
ATARI INCORPORATED		BRODERBUND SOFTWARE	
<input type="checkbox"/> Word Processor (d)	149.95	<input type="checkbox"/> Apple Panic (d)	29.95
<input type="checkbox"/> Personal Finance Management (d)	74.95		21.00
<input type="checkbox"/> Caverns of Mars (d)	39.95		
<input type="checkbox"/> Invitation to Programming 2 (t)	29.95		
<input type="checkbox"/> Invitation to Programming 3 (t)	29.95		
<input type="checkbox"/> Conversational French (t)	59.95		
<input type="checkbox"/> Conversational German (t)	59.95		
<input type="checkbox"/> Conversational Italian (t)	59.95		
<input type="checkbox"/> Conversational Spanish (t)	59.95		
<input type="checkbox"/> The Programmer Kit (c)	79.85		
<input type="checkbox"/> The Entertainer Kit (c)	116.75		
<input type="checkbox"/> Asteroids (c)	44.95		
<input type="checkbox"/> Assembler Editor (c)	59.95		
<input type="checkbox"/> Atari Basic (c)	59.95		
<input type="checkbox"/> Space Invaders (c)	44.95		
<input type="checkbox"/> Missile Command (c)	44.95		
<input type="checkbox"/> Star Raiders (c)	49.95		
<input type="checkbox"/> Centipede (c)	44.95		
<input type="checkbox"/> Pac man (c)	44.95		
ARCADE PLUS		CAVALIER	
<input type="checkbox"/> Ghost Hunter (d)	34.95	<input type="checkbox"/> Bug Attack (d) (t)	29.95
<input type="checkbox"/> Ghost Hunter (t)	29.95		21.00
ARTSCI		COMPUTER MAGIC, LTD.	
<input type="checkbox"/> Poker Solitaire (d)	24.95	<input type="checkbox"/> Kayos (d) (t)	34.95
<input type="checkbox"/> Gomoku (d)	29.95		25.00
<input type="checkbox"/> Reversi (d)	29.95		
<input type="checkbox"/> Poker Solitaire (t)	14.95		
<input type="checkbox"/> Gomoku (t)	19.95		
<input type="checkbox"/> Reversi (t)	19.95		
		DATASOFT	
		<input type="checkbox"/> Atari Mailing Lis (d)	24.95
		<input type="checkbox"/> Text Wizard (d)	99.95
		<input type="checkbox"/> Dung Beetle (d)	29.95
			21.00
		GEBELLI	
		<input type="checkbox"/> Match Racers (d)	29.95
		<input type="checkbox"/> Andromeda (d)	39.95
		<input type="checkbox"/> Pathfinder (d)	34.95
			25.00
		INNOVATIVE DESIGN	
		<input type="checkbox"/> Pool 1.5 (d)	34.95
			25.00
		JV SOFTWARE	
		<input type="checkbox"/> Action Quest (d) (t)	29.95
			21.00
		K-BYTE	
		<input type="checkbox"/> K-Razy Shootout (c)	49.95
			36.00
		ON-LINE SYSTEMS	
		<input type="checkbox"/> Crossfire (d) (t)	29.95
		<input type="checkbox"/> Mousekattack (d)	34.95
		<input type="checkbox"/> Jawbreaker (d) (t)	29.95
		<input type="checkbox"/> Threshold (d)	39.95
		<input type="checkbox"/> Frogger (d)	34.95
			25.00
		SYNERGISTIC	
		<input type="checkbox"/> Warlock's Revenge (d)	35.00
			25.00

WE DEAL IN BUSINESS•EDUCATION•ENTERTAINMENT SOFTWARE

ATARI JOYSTICKS

SANYO 13 INCH

Color Monitor

ZENITH ZVM-121

Green Monitor

VERBATIM DATALIFE

Box of 10 Disks

\$8

\$425

\$120

\$26

CREDIT CARD ORDERS ACCEPTED BY PHONE 4 PM-7 PM DAILY MONDAY THRU FRIDAY

<input type="checkbox"/> PLEASE SEND THE PROGRAMS CHECKED	
<input type="checkbox"/> PLEASE SEND THE ACCESSORIES CHECKED	
<input type="checkbox"/> PLEASE SEND FREE CATALOG	DRIVE — <input type="checkbox"/> DISK <input type="checkbox"/> CASSETTE
NAME _____	
ADDRESS _____	
STATE _____ ZIP _____	AMT. ENCLOSED _____
CHARGE MY <input type="checkbox"/> VISA <input type="checkbox"/> MASTERCARD # _____	
EXPIRES _____	SIGNATURE _____

■ MFGS. TRADEMARK

MAIL TO **STRÖM** SYSTEMS INC.
P.O. Box 197
Plymouth, Mi. 48170
(313) 455-8022

Send check or money order for total purchase price, plus \$2. postage handling. Michigan Residents add 4% tax - C.O.D. add \$5.

MIDNIGHT STRIP

By M. L. CLAYTON

This program requires 24 K RAM system for cassette and a 32K RAM system for disk. It is a simulation of a trip down an Obstacle race course as viewed from the drivers seat. It uses either paddle or joystick input and is independent of keyboard interaction.

There has been a small attempt at program documentation due to the limited space available in M.A.C.E. magazine. The code has been somewhat optimized for a balance of speed and flash in a minimum of memory, therefore it appears that the listing is all squeezed together; this arrangement can be altered. Many things can be altered or modified such as the creation of a braking routine (joystick pull back or unused paddle trigger) or passing objects (trees or cars), using the unused player, or converting the work loop into machine. With the limited time available to me, I've attempted to create a workable (and fun to play) program for beginning and intermediate BASIC programmers to be understood and tailored to each users likes.

This program was originally designed for paddle input. Joystick input was added as an afterthought, so to keep play consistant for both controllers I've made the joystick simulate the paddle. The maximum possible score is 10,000 which is a feat only possible with the paddles. I would be interested in knowing the joystick high score (no modifications please).

When using the paddle controllers, you should keep the unused controller turned fully clockwise to insure that the computer will correctly sense that the paddles are connected.

Attach your controller to the left port and
HAVE A NICE DRIVE!

Program Documentation for
Midnight Strip
(c) 1982 all rights reserved
M.L. Clayton
45655 Utica Gr. West
Utica, MI 48087

INITIALIZATION & TITLE

LINE 3 -- DIM arrays & strings. Turn off display for faster inits.

1200-1210 -- Variable initializations and game start ditty generation.

1211 -- Fill memory routine in string form.

1212-1226 -- More variable inits and game start ditty generation.

1050-1057 -- Calculate 3K below ramtop location for PMG base address. Set up title page player attributes. Restart DMA.

1060-1085 Set up string location pointers to point to PMG ram start.

1093-1095 -- Set up title playfield. Kill attract mode.

1096 -- Clears PMG RAM.

1097-1100 -- Assign title page player data. Turn on PMG.

1101 - Determine if paddle or joystick.

1102-1109 -- Allows user input of track type with visual and audio feedback of selection. Attract selection tone generation after about 30 seconds.

1110 -- Put up start engine message.

1111 -- Determine if standard track already exists in memory.

1112-1115 -- If necessary assign standard or random values. Create engine cranking sound.

1116 -- Create engine starting sound.

1117 -- Create idleing engine sound. Initialize track position and race timer to zero. Kill attract mode again.

MAIN SCREEN DISPLAY

73-83 -- Set up and activate the five Display List Interrupts.

84 -- Determine start of screen ram. Calculate windshield bound bytes. Calculate rear view mirror reflection bytes. Initialize vehicle velocity and speedometer end position.

85-86 -- Initialize text window contents (speedometer numbers).

87-88 -- Draw sun visors.

89 -- Draw mirror.

90-100 -- Draw lower part of screen. (brake lights, indicator, hood)

401 -- Force user to center paddles before starting. Sets ready to start indicator red and loops until centered.

403 -- Sets indicator to green. Continue loop until centered.

404 -- Trigger pressed; time to start. Turn off indicator and center joystick position.

405 Clears PMG RAM.

406 --Assigns PMBASE. Set pylon count indicator color.

407 -- Sets horizontal position of pylon

continued

numbers.

409 Sets quad width players and player priority.

410 Starts work loop.

MAIN WORK LOOP

LINE 6 -- Read paddle position.

7 -- Compute new joystick position from old position and current joystick orientation.

9 -- Compute current velocity. If not pushing trigger then decrease velocity.

10 -- Keep velocity within bounds.

11 -- If not pushing trigger then display brake lights. Move speedometer indicator to left.

12 -- If trigger is pressed then kill brake lights. Increment speedometer to right.

13-14 -- Evaluate new current speedometer end byte. Sound engine. Determine new track position and fractional track length.

15 -- Assign correct track color.

16-32 -- Assign centered pylon positions; and scale factors.

33-34 -- Offset the data to target positions.

47-57 -- Place any pylon data that wanders out of the windshield area into rear view mirror position.

58 -- Restore mirror edges to default values. Clear windshield RAM.

60-64 -- Draw road.

65-69 -- Draw pylons.

70 -- At end blank out last pylon set. Turn off pylon number indicatr.

72 -- Set horizontal player positions to zero. Adjust sound.

73 -- Increment timer if no crash occurs. Re-enter the main loop.

74 -- If crashed move pylon number indicators off screen. Go to crash routine.

CRASH ROUTINE

500 -- Make crash sound.

501 -- Rotates colors.

502 -- More crash sounds.

503 -- Draw cracks.

504 -- More color rotations.

505 -- Clear PMG RAM.

507 -- Diminish crash sounds.

508 -- Creates crash information panel. Turn off all sounds.

509 -- Create final velocity display.

511 -- Display track position message.

512 -- Calculate score. Jump to score display routine.

MISCELLANEOUS

600-603 -- Title display DATA

900-915 -- Race course DATA.

1000 -- Delay List Interrupt skeleton DATA

SUCCESSFUL COMPLETION (END) ROUTINE

3000-3001 -- Brighten then dim display. Decrease sound to off.

3002-3004 Calculate score and ranking pointer.

3005-3006 -- Create finish strip display.

3008-3012 -- Generate score display. Jump to audio-visual display routine if excellent score. Delay then put up options.

3013 -- Monitor trigger and restart when pressed.

3016 -- Rotate high score message.

3017-3022 -- Determine if paddle or stick score.

4000-4001 -- Excellent score audio-visual routine.

5000 -- RAM FILL SUBROUTINE.

Note: FS\$ in line 1211 will not copy to the printer, so you must build that string from the following DATA statements, then enter it in line 1211.

TO BUILD FS\$ FOR MIDNIGHT STRIP

```
FOR I=1 TO 56:READ DATA
```

```
FS$(LEN(FS$)+1)=CHR$(DATA)
```

```
NEXT I
```

```
DATA 104,104,133,209,104,133
```

```
DATA 208,104,141,255,6,104,141
```

```
DATA 254,6,104,104,168,162,0
```

```
DATA 152,129,208,56,173,254,6
```

```
DATA 229,208,208,8,173,255,6
```

```
DATA 229,209,208,1,96,24,169
```

```
DATA 1,101,208,133,208,169,0
```

```
DATA 101,209,133,209,169,0,240,220
```



```

0 REM XXX XXX
1 REM XXX BY M. L. Clayton XXX
2 REM XXX Copyright 1982 XXX
3 DIM CP$(102),SN1$(8),SN2$(8),SN3$(8)
,NUST$(182),MESS$(54),R(262),FS$(58),RC(3):POKE
559,0:GOSUB 1200
4 GOSUB 1050
5 GOTO 75
6 IF TR=636 THEN PAD=PEEK(P0)
7 PU=PEEK(TR):IF TR=644 AND PEEK(SK)<13 AND VEL>0 THEN
PAD=PAD-22*SGN(NB-PEEK(SK))
8 OFF=(PAD-114)/3.3:A=(N1-PU)*N4-N2:VEL=VEL+A*1.5:IF
PU THEN VEL=VEL-N2:COLOR Z
9 IF VEL<Z THEN VEL=Z
10 IF VEL>120 THEN VEL=120
11 IF PU THEN POKE 1638,50:PLOT ENOM,N38:IF ENOM<77
THEN PLOT ENOM+N1,N38:DRAINTO ENOM+N3,N38
12 IF PU=Z THEN POKE 1638,Z:COLOR N3:PLOT ENOM,N38:IF
ENOM<SN THEN PLOT ENOM+N1,N38:V=VEL/20
13 ENOM=INT(VEL*GT/301):SOUND Z,50,N4,V*N2+N2:SOUND
N 2 , G T + N 3 2 , N 2 , V * 2 . 5 : S O U N D
N3,60,N6,V*2.5:B=VEL/N50:BU=INT(B)
14 RC(N1)=N2:RC(N2)=N2:A=INT
((B-BU)*N8)+N1:IF VEL>N60 THEN SOUND
N1,76-INT((VEL-N60)/N3)*N2,N2,INT((VEL-N60)/N4)
15 RC(N3)=N2:BU=BU-INT(BU/HU)
*HU:BU=BU-INT(BU/N10)*N10:IF BU>0 AND BU<4 THEN
RC(BU)=N1
16 IF A=N1 THEN Y1L=N8:Y1AL=N8:
Y2L=N10:Y2AL=11:Y3L=13:Y3AL=15
:Y4L=17:Y4AL=20:Y5L=21:Y5AL=26
:J1=2.5:J2=2.6:J3=2.76
17 IF A=N1 THEN L1=N38:R1=42:L2=36
:R2=44:L3=N32:R3=48:L4=25:R4=55
:L5=16:R5=64:J4=2.95:J5=3.3:GOTO N32
18 IF A=N2 THEN Y1L=N8:Y1AL=N8:Y2L=10
:Y2AL=11:Y3L=13:Y3AL=15:Y4L=17:Y4AL=21
:Y5L=21:Y5AL=27:J1=2.52:J2=2.62:J3=2.78
19 IF A=N2 THEN L1=N38:R1=42:L2=36
:R2=44:L3=32:R3=48:L4=24:R4=56:L5=15
:R5=65:J4=2.98:J5=3.35:GOTO N32
20 IF A=N3 THEN Y1L=N9:Y1AL=N9:Y2L=11
:Y2AL=12:Y3L=13:Y3AL=16:Y4L=18:Y4AL=22
:Y5L=22:Y5AL=27:J1=2.53:J2=2.64:J3=2.81
21 IF A=N3 THEN L1=N38:R1=42:L2=35
:R2=45:L3=31:R3=49:L4=23:R4=57:L5=13
:R5=67:J4=3.02:J5=3.4:GOTO N32
22 IF A=N4 THEN Y1L=N9:Y1AL=N9:Y2L=11
:Y2AL=12:Y3L=14:Y3AL=17:Y4L=18:Y4AL=22
:Y5L=23:Y5AL=27:J1=2.54:J2=2.66:J3=2.83
23 IF A=N4 THEN L1=N38:R1=42:L2=35
:R2=45:L3=30:R3=50:L4=22:R4=58:L5=12
:R5=68:J4=3.06:J5=3.45:GOTO N32
24 IF A=N5 THEN Y1L=N9:Y1AL=N9:Y2L=12
:Y2AL=13:Y3L=15:Y3AL=18:Y4L=19:Y4AL=23
:Y5L=24:Y5AL=27:J1=2.56:J2=2.68:J3=2.85

```

```

25 IF A=N5 THEN L1=37:R1=43:L2=34:R2=46
:L3=29:R3=51:L4=21:R4=59:L5=10:R5=70
:J4=3.1:J5=3.5:GOTO N32
26 IF A=N6 THEN Y1L=N9:Y1AL=N9:Y2L=12
:Y2AL=13:Y3L=15:Y3AL=18:Y4L=19:Y4AL=23
:Y5L=25:Y5AL=27:J1=2.57:J2=2.72:J3=2.9
27 IF A=N6 THEN L1=37:R1=43:L2=34:R2=46
:L3=28:R3=52:L4=20:R4=60:L5=9:R5=71
:J4=3.15:J5=3.6:GOTO N32
28 IF A=N7 THEN Y1L=N10:Y1AL=N10:Y2L=12
:Y2AL=14:Y3L=16:Y3AL=19:Y4L=20:Y4AL=24
:Y5L=26:Y5AL=27:J1=2.58:J2=2.72
29 IF A=N7 THEN L1=37:R1=43:L2=33:R2=47
:L3=27:R3=53:L4=19:R4=61:L5=7:R5=73
:J3=2.9:J4=3.2:J5=3.7:GOTO N32
30 Y1L=N10:Y1AL=N10:Y2L=12:Y2AL=14
:Y3L=16:Y3AL=19:Y4L=20:Y4AL=25:Y5L=27
:Y5AL=27:J1=2.59:J2=2.74:J3=2.92:J4=3.25
31 L1=37:R1=43:L2=N32:R2=48:L3=26
:R3=54:L4=18:R4=62:L5=N4:R5=76:J5=3.8
32 Y1R=Y1L:Y1AR=Y1AL:Y2R=Y2L:Y2AR=Y2AL
:Y3R=Y3L:Y3AR=Y3AL:Y4R=Y4L:Y4AR=Y4AL
:Y5R=Y5L:Y5AR=Y5AL
33 L1=L1+OFF+R(B+N4)*J1:R1=R1+OFF+R
(B+N4)*J1:L2=L2+OFF+R(B+N3)*J2:R2=R2
+OFF+R(B+N3)*J2:L3=L3+OFF+R(B+N2)*J3
34 R3=R3+OFF+R(B+N2)*J3:L4=L4+OFF+R
(B+N1)*J4:R4=R4+OFF+R(B+N1)*J4:L5=L5
+OFF+R(B)*J5:R5=R5+OFF+R(B)*J5
35 SN1=INT(B/HU):SN2=INT((B-SN1)*HU)
/N10):SN3=INT(B-SN1*HU-SN2*N10)
36 SN1$=NUST$(SN1*N8+N1,SN1*N8+N8)
:SN2$=NUST$(SN2*N8+N1,SN2*N8+N8)
:SN3$=NUST$(SN3*N8+N1,SN3*N8+N8)
47 IF L1<Z OR L1>SN THEN Y1L=N6:Y1AL=N6
:L1=39+N8*SGN(L1)
48 IF L2<Z OR L2>SN THEN Y2L=N6:Y2AL=N6
:L2=39+N8*SGN(L2)
49 IF L3<Z OR L3>SN THEN Y3L=N6:Y3AL=N6
:L3=39+N8*SGN(L3)
50 IF L4<Z OR L4>SN THEN Y4L=N6:Y4AL=N6
:L4=39+N8*SGN(L4)
51 IF L5<Z OR L5>SN THEN Y5L=N6:Y5AL=N6
:L5=39+N8*SGN(L5)
53 IF R1>SN OR R1<Z THEN Y1R=N6:Y1AR=N6
:R1=39+N8*SGN(R1)
54 IF R2>SN OR R2<Z THEN Y2R=N6:Y2AR=N6
:R2=39+N8*SGN(R2)
55 IF R3>SN OR R3<Z THEN Y3R=N6:Y3AR=N6
:R3=39+N8*SGN(R3)
56 IF R4>SN OR R4<Z THEN Y4R=N6:Y4AR=N6
:R4=39+N8*SGN(R4)
57 IF R5>SN OR R5<Z THEN Y5R=N6:Y5AR=N6
:R5=39+N8*SGN(R5)
58 POKE ML,12:POKE MR,Z:FILL=USR
(ADR(FS$),MEM1,Mem2,Z)
60 COLOR RC(N1):IF Y1L>N6 AND Y2L>N6 AND B<252 THEN

```

continued

```

PLOT L1,Y1AL:DRAHTO
L2,Y2AL
61 IF Y1R>N6 AND Y2R>N6 AND B<252 THEN PLOT
R1,Y1AR:DRAHTO R2,Y2AR
62 COLOR RC(N2):IF Y2L>N6 AND Y3L>N6 AND B<253 THEN
PLOT L2,Y2AL
:DRAHTO L3,Y3AL
63 IF Y2R>N6 AND Y3R>N6 AND B<253 THEN PLOT
R2,Y2AR:DRAHTO R3,Y3AR
64 COLOR RC(N3):IF Y3L>N6 AND Y4L>N6 AND B<254 THEN
PLOT L3,Y3AL:DRAHTO L4,Y4AL
65 IF Y3R>N6 AND Y4R>N6 AND B<254 THEN PLOT
R3,Y3AR:DRAHTO R4,Y4AR
66 COLOR N1:PLOT L5,Y5L:DRAHTO L5,Y5AL:PLOT
R5,Y5R:DRAHTO R5,Y5AR:IF B<252 THEN PLOT L1,Y1L:PLOT
R1,Y1R
67 IF B<253 THEN PLOT L2,Y2L:DRAHTO L2,Y2AL:PLOT
R2,Y2R:DRAHTO R2,Y2AR
68 IF B>TF THEN PLOT L4,Y4L:DRAHTO L4,Y4AL:PLOT
R4,Y4R:DRAHTO R4,Y4AR
69 IF B<254 THEN PLOT L3,Y3L:DRAHTO L3,Y3AL:PLOT
R3,Y3R:DRAHTO R3,Y3AR
70 IF B>=TF THEN SOUND N1,N50,N10,N10:COLOR Z:PLOT
L5,Y5L:DRAHTO L5,Y5AL:PLOT R5,Y5R:DRAHTO R5,Y5AR
71 IF B>=TF THEN POKE 1613,Z:POKE SH+N1,Z:POKE
SH+N2,Z:POKE SH+N3,Z:FOR I=FTT TO FTT+N2:POKE I,Z:NEXT
I
72 IF B>=TF THEN FOR I=N1 TO N5*N6:NEXT I:SOUND
N1,Z,Z:LM=PMBXE:UM=LM+TTT:FI=Z:GOSUB FG:GOTO 3000
73 T=T+N1:IF L5<35 AND R5>46 THEN GOTO N6
74 POKE 1613,Z:POKE SH+N1,Z:POKE SH+N2,Z:POKE
SH+N3,Z:POKE FTT,Z:POKE FTT+N1,Z:POKE FTT+N2,Z:GOTO
FG/N10
75 RESTORE HU*N5*N2:FOR I=Z TO N24
:READ N:POKE 1536+I,N:POKE 1561+I,N
:POKE 1586+I,N:POKE 1611+I,N
:POKE 1636+I,N
76 NEXT I:POKE 1563,198:POKE 1575,N50
:POKE 1588,82:POKE 1600,75:POKE 1625,HU
77 POKE 1650,Z:POKE 1572,22
:POKE 1597,24:POKE 1622,23:POKE 1647
,24:FOR I=SH TO 707:POKE I,Z:POKE 52544
+I,Z:NEXT I
78 GRAPHICS N2*N10+N1:POKE SH+N4,N10
:POKE SH+N5,224:POKE SH+N6,148
:POKE SH+N8,Z
79 DL=PEEK(560)+EXPEEK(561)
80 POKE DL+N32,138:POKE DL+41,138:POKE DL+42,134
:POKE DL+44,138:POKE DL+45,138
81 FOR I=N1 TO N3:POKE DL+N50+N1+I,
PEEK(DL+N50+N2+I):NEXT I
83 POKE 512,Z:POKE 513,N6:POKE 54286,192
84 SRS=PEEK(DL+N4)+EXPEEK(DL+N5)
:MEM1=SRS+N8*N2*N10:MEM2=SRS+559
:ML=SRS+127:MR=ML+N4:VEL=Z:ENOM=Z
85 SD=SRS+740:POKE SD,N4*N4:POKE SD+N4+N5,22
,19:POKE SD+N5,N4*N4:POKE SD+N4+N5,22
:DPOKE SD+N5+N5,N4*N4
86 POKE SD+14,25:POKE SD+15,16:POKE SD
+17,17:POKE SD+18,18:POKE SD+19,16
87 COLOR N2:PLOT 29,N4:DRAHTO 33,Z:DRAHTO
N1,Z:POSITION N5,N4:POKE
765,N2:XIO N3*N6,$6,Z,Z,"S":PLOT SN,N9*N4
88 PLOT 73,N4:DRAHTO 77,Z:DRAHTO 45,Z:POSITION
49,N4:XIO 18,$6,Z,Z,"S"
89 COLOR N3:PLOT N39,Z:DRAHTO N39,N3
:DRAHTO 31,N3:DRAHTO 30,N7:PLOT
40,N3:DRAHTO 47,N3:DRAHTO 48,N7:DRAHTO 30,N7
90 COLOR N2:PLOT Z,35:DRAHTO N5,32:DRAHTO 33,29:DRAHTO
46,29:DRAHTO 74,32:DRAHTO 79,35:DRAHTO Z,35:DRAHTO Z,36
91 PLOT 39,28:PLOT 40,28:PLOT 79,36:PLOT N3,34:DRAHTO
76,34:PLOT 74,33:DRAHTO 5,33:PLOT N10,32:DRAHTO 69,N32
92 PLOT 19,31:DRAHTO 59,31:PLOT 29,30:DRAHTO N50,30
93 COLOR Z:PLOT 39,29:DRAHTO 39,34:DRAHTO 40,34:DRAHTO
40,29:POKE 1613,Z:POKE 1638,Z
94 COLOR N2:PLOT N38,46:DRAHTO N38,42:DRAHTO
40,41:DRAHTO 41,42:DRAHTO 41,46
95 COLOR N3:PLOT N9,43:PLOT N9,44:PLOT N10,43:PLOT
N10,44
96 PLOT 13,44:PLOT 13,45:PLOT 14,44:PLOT 14,45
97 PLOT 17,43:PLOT 17,44:PLOT 18,43:PLOT 18,44
98 PLOT 60,43:PLOT 60,44:PLOT 61,43:PLOT 61,44
99 PLOT 64,44:PLOT 64,45:PLOT 65,44:PLOT 65,45
100 PLOT 68,43:PLOT 68,44:PLOT 69,43:PLOT 69,44
101 IF TR=PT THEN POS=PEEK(PO):IF POS<76 OR POS>153
THEN POKE 1613,N50:GOTO 401
103 POKE 1613,210:IF PEEK(TR)>Z THEN GOTO 401
104 POKE 1613,Z:PAD=HU+N7+N7+R(Z)*N6*N3
105 LM=PMBXE:UM=LM+TTT:FI=Z:GOSUB FG
106 POKE 559,46:FOR I=SH+N1 TO SH+N3:POKE I,230:NEXT I
107 POKE FTT,86:POKE FTT+N1,111:POKE FTT+N2,136
109 FOR I=FTT+N8 TO FTT+N10:POKE I,N3:NEXT I:POKE
FTT+N24+N4,N3
110 GOTO N6
500 R1=N3*N3+INT(RND(Z)*17):R2=30+INT
(RND(Z)*20):COLOR N3:FOR I=N5*N5*N3
TO 45 STEP -N2:SOUND Z,N4*N5,N4*N2,I/N5
501 POKE SH+N4,RND(Z)*TF:POKE SH+N5,
RND(Z)*TF:SOUND N1,N3*N5*N3,N4+N4,
I/N5+N3:SOUND N2,HU+N5*N5*N3,N4+N4,I/N5
502 SOUND N3,TF,N4+N4,I/N5+N5:POKE 1538
,RND(Z)*TF:POKE 1563,RND(Z)*TF
503 A=INT(RND(Z)*SN):V=9+INT(RND(Z)*17)
:PLOT R2,R1:DRAHTO A,V
504 POKE 1588,RND(Z)*TF:POKE 1638
,RND(Z)*TF:NEXT I:POKE 559,Z
505 LM=PMBXE:UM=LM+TTT:FI=Z:GOSUB FG
507 FOR I=N50 TO Z STEP -N2:SOUND Z,58,
N8,I/N3:SOUND N1,GT-N8,N3*N4,I/N3:NEXT I
508 GRAPHICS N3*N6:POSITION N1,Z:?"$6";
"WE HAVE CONTACT !!"":FOR I=Z TO N3
:SOUND I,Z,Z:NEXT I
509 POSITION Z,N2:?"$6";
";INT(VEL);"

```

```

511 POSITION N1,N4? #6;" PYLON ";INT(B)
512 OFF=INT(B/E*FG*N2+(125*(B/E)-T)
  *N7*N8+INT(VEL)):IF OFF<Z THEN OFF=Z
513 L1=Z:GOTO 3008
600 DATA 205,181,165,181,205,1,0,14,
  10,14,2,2,2,0,28,20,28,20,28,0,112,
  16,16,112,64,120,0,0
601 DATA 255,219,219,219,195,195,195,
  231,0,224,192,192,192,192,192,193,255,
  0,255,193
602 DATA 192,192,192,192,193,255,0,16
  ,16,16,16,16,0,120,72,8,120,72,120
  ,8,0,68,68
603 DATA 124,8,8,8,56,0,16,120,16,16
  ,16,16,24,0,124,68,68,68,68,68,124,0
  ,64,68,100,84,76,68,68,
900 DATA 0,0,0,0,0,1,2,3,4,5,6,7,9,
  10,8,7
901 DATA 5,4,3,2,1,0,-1,-2,-3,-4,-3,-2,
  -1,-1,0,1
902 DATA 0,1,2,3,4,5,7,8,9,11,13,14,15,
  15,13,12
903 DATA 10,9,8,6,5,3,1,-1,-2,-3,-3,-2,
  -1,0,0,1
904 DATA 1,2,2,1,0,2,4,3,1,0,-2,-4,-6,
  -7,-8,-10
905 DATA -11,-12,-14,-15,-15,-14,-13,
  -15,-14,-12,-10,-8,-6,-4,-2,-1
906 DATA 0,2,3,5,6,7,8,9,10,9,10,11,
  12,14,14,15
907 DATA 14,15,16,16,15,13,12,10,9,7,6,
  5,4,3,2,1
908 DATA -1,-2,-4,-5,-7,-8,-9,-9,-8,-8,
  -8,-9,-9,-8,-7,-6
909 DATA -5,-4,-2,-0,-1,-2,-3,-3,-2,
  -1,-1,0,-1,-2,-2,-1
910 DATA 0,1,2,3,4,4,5,6,7,8,9,8,6,5,
  3,3
911 DATA 3,3,5,6,7,8,9,8,9,10,9,7,
  6,5,3,1
912 DATA 1,2,1,0,-1,-2,-3,-4,-6,-7,-8,
  -9,-10,-12,-11,-11
913 DATA -10,-10,-9,-8,-7,-6,-6,-7,-6,
  -5,-4,-4,-3,-2,-1,0
914 DATA -1,-3,-2,-1,1,0,2,3,4,5,4,4,
  5,5,4,4
915 DATA 4,2,1,0,-1,-3,-4,-5,-5,-5,
  -4,-3,-2,-1,1,0
1000 DATA 72,169,2,69,79,37,78,141,10,
  212,141,23,208,169,25,141,0,2,169,6,
  141,1,2,104,64
1050 GRAPHICS N3*N6
1055 PMB=PEEK(106)-N6-N6:POKE 559,46
  :POKE 54279,PMB:POKE 623,N1
1056 POKE SH,HU+N2:POKE SH+N1,148:POKE SH
  +N2,N6*N3:POKE SH+N3,N2
1057 POKE 53256,N1:POKE 53257,N3:POKE 53258,N3:POKE

```

```

53259,N3
1060 VT=PEEK(134)+PEEK(135)*E
1065 AT=PEEK(140)+PEEK(141)*E
1070 VEL=PMB*E+524-AT:POKE VT+N3,INT(VEL/E):POKE
  VT+N2,VEL-PEEK(VT+N3)*E
1075 R1=VEL+217:R2=R1+128:R3=R2+128
1080 POKE VT+N5+N5,R1-INT(R1/E)*E:POKE
  VT+N5+N6,INT(R1/E)
1081 POKE VT+18,R2-INT(R2/E)*E:POKE VT+19,INT(R2/E)
1082 POKE VT+26,R3-INT(R3/E)*E:POKE VT+27,INT(R3/E)
1085 POKE FTT-N1,188:POKE FTT,15:POKE
  FTT+N1,HU+N2:POKE FTT+N2,183
1093 POKE SH+N5,66:POKE SH+N7,194:POKE 77,Z:POSITION
  Z,Z? #6;"":POSITION N2,N1
1094 POKE SH+N4,234? #6;" MIDNIGHT",,"      STRIP
  ":"POSITION Z,N3? #6;"""
1095 POSITION N1,N5? #6;"      random ",,""
1096 LM=PMB*E:UM=LM+1023:FI=Z:GOSUB FG
1097 CP$=NUST$(N9*N9,182):DL=PMB*E
  :LM=DL+658:POKE LM,N3:POKE LM+N3,N3
  :POKE LM+N24,N3:POKE LM+N24+N3,N3
1098 LM=DL+768:UM=LM+N8*N2:FI=170:GOSUB FG
  :LM=LM+47:UM=LM+N8:GOSUB FG
1099 LM=LM+33:UM=LM+N7:GOSUB FG
  :LM=LM+N24:UM=LM+N7+N8:GOSUB FG
1100 LM=PMB*E+896:UM=PMB*E+TTT
  :FI=TF:GOSUB FG:POKE 53277,N3
1101 TR=PT:JUMP=1102:IF PEEK(PO)
  +PEEK(PO+N1)=456 THEN TR=ST
  :JUMP=1104:GOTO JUMP
1102 IF PEEK(PO)>114 THEN POKE SH+N5,76
  :POKE SH+N7,194:CH=Z:GOTO 1106
1103 CH=N1:POKE SH+N7,204:POKE SH+N5,
  66:GOTO 1106
1104 IF PEEK(SK)=N6+N7 THEN POKE SH+N5
  ,76:POKE SH+N7,194:CH=Z:GOTO 1106
1105 CH=N1:POKE SH+N7,GT+N4:POKE SH+N5,66
1106 IF PEEK(77)<N1 THEN SOUND
  Z,121-CH*61,N10,N2:GOTO 1109
1107 SOUND Z,Z,Z,Z:GOTO 1109
1109 POKE SH,RND(Z)*255:IF PEEK(TR) THEN GOTO JUMP
1110 POSITION N4,N4+N5? #6;" YOUR ",,"":SOUND
  Z,Z,Z,Z:FOR I=N1 TO HU:NEXT I
1111 IF SRS>Z AND CH THEN
  A=R(N3)+R(56)+R(77)+R(156)+R(N2*HU):IF A=-N4*N4 THEN
  GOTO 1116
1112 IF CH=Z THEN V=-N1:GOTO 1114
1113 RESTORE 900:FOR I=Z TO TF:READ N:R(I)=N:SOUND
  Z,ABS(N*(N7+N8)),N8,N4:NEXT I:GOTO 1116
1114 FOR I=Z TO N6+N1:RESTORE
  INT(RND(Z)*16/N2)*N2+900:FOR CH=Z TO 31:V=V+N1:READ
  N:R(V)=N:SOUND Z,ABS(N*15),N8,N4
1115 NEXT CH:NEXT I
1116 A=Z:FOR I=Z TO N32-N1:A=N1-A:SOUND
  Z,N50,A,(N3*N5)-INT(I/N3):NEXT I
1117 SOUND Z,N50,N4,N2:SOUND N1,GT,N6,N4:B=Z:T=Z:POKE
  T,Z:RETURN

```

continued

```

1200 Z=0:N1=1:N8=8:N10=10:N50=50
;HU=100:FG=N50*HU:SH=704:TF=255
;TJ=3013:SOUND Z,243,N10,N10:GOSUB 1214
1201 FOR I=N1 TO 80:NEXT I:SN=79
;GT=200:SK=632:PT=636:ST=644:P0=624
;E=256:FTT=53249
1202 SOUND Z,136,N10,N10:FOR I=N1 TO N50:NEXT I:GOSUB
1214
1205 N2=2:N3=3:N4=4:N5=5:N6=6:N7=7
;N8=8:N9=9:N24=24:N32=32:N38=38:N39=39
;N60=60:TTT=1023:HI=Z:OT=Z
1207 SOUND Z,121,N10,N10:FOR I=N1 TO HU:NEXT I:GOSUB
1214
1210 MESS$="TRY AGAIN PATHETIC! FINISHED! GOOD
JOB! EXCELLENT AMAZING!!"
1211 F$=" "(see text)
1212 SOUND Z,162,N10,N10:FOR I=N1 TO N50:NEXT I:GOSUB
1214:SOUND N1,HU+N2,N10,N8:GOTO 1215
1214 POKE SH+N8,RND(Z)*TF:RETURN
1215 L1=PEEK(756)*E:FOR I=N1 TO
80:NUST$(I,I)=CHR$(PEEK(L1+127+I))
1216 IF I=30 THEN SOUND N3,91,N10,N10:GOSUB 1214
1217 IF I=70 THEN FOR A=Z TO 15 STEP N3:FOR N=Z TO E
STEP N3:SOUND N3,N,N10,A:NEXT N:NEXT A
1218 NEXT I:SOUND Z,136,N10,N10:GOSUB 1214
1220 FOR I=N1 TO HU+N2:READ
N:NUST$(I+80,I+80)=CHR$(N):IF I=30 THEN SOUND
N3,136,N10,N7*N2:GOSUB 1214
1221 IF I=45 THEN SOUND N3,91,N10,N10:GOSUB 1214
1222 IF I=60 THEN SOUND N3,243,N10,N7*N2:GOSUB 1214
1223 IF I=77 THEN SOUND N3,45,N10,N8:GOSUB 1214
1224 IF I=HU THEN SOUND N3,Z,Z,Z:GOSUB 1214
1225 NEXT I:SOUND N1,121,N10,N10:FOR I=N1 TO N50:NEXT
I:SOUND Z,243,N10,N8:FOR I=N1 TO N50:NEXT I:POKE
SH+N8,Z
1226 FOR I=243 TO 60 STEP -N1:SOUND N1,I,N10,N10:NEXT
I:SOUND Z,Z,Z:Z:SOUND N1,Z,Z,Z:RETURN
3000 SOUND N2,Z,Z,Z:FOR I=INT(VEL/N8) TO Z STEP
-N1:POKE SH+N4,I:POKE SH+N5,I:POKE SH+N6,I:POKE
1563,I:POKE 1588,I
3001 POKE 1613,I:POKE 1538,I:POKE 1638,I:SOUND
Z,N50,N4,I:SOUND N3,GT,N6,I:FOR A=Z TO N6*N3:NEXT
A:NEXT I
3002 FOR I=Z TO N50:NEXT
I:OFF=FG*N2-HU-N10*N2+(125-T)*N7*N8+INT(VEL):IF OFF<Z
THEN OFF=Z
3003 ENOM=INT(ABS(OFF+N50)/2000):IF OFF>7999 AND
OFF<9000 THEN ENOM=ENOM-N1
3004 L1=Z:IF OFF>9500 THEN L1=N1:IF OFF>9700 THEN
L1=N2:IF OFF>9800 THEN L1=N3:IF OFF>9900 THEN L1=N4
3005 GRAPHICS N6*N3:FOR I=Z TO N3:SOUND I,Z,Z,Z:NEXT
I:IF OFF>Z THEN POSITION N5,Z:?
;MESS$(ENOM*9+N1,ENOM*9+9)
3006 POSITION Z,N2:?
;INT(VEL);"
3008 POSITION N1,N5:?
;OFF:IF L1>Z
THEN GOSUB FG-HU*N10

```

```

3009 IF TR=ST THEN POSITION N5,N7:?
;6;"(JOYSTICK)":GOTO 3011
3010 POSITION N6,N7:?
;"(PADDLE)"
3011 FOR I=Z TO E:NEXT I:POSITION N2,N10:?
;6;""
;POSITION Z,N3*N3
3012 ? ;6;"":A=Z:IF HI<OFF THEN HI=OFF:OT=TR
3013 FOR I=N1 TO HU:IF PEEK(PT)=Z OR PEEK(ST)=Z THEN 4
3014 NEXT I
3016 A=N1-A:IF A=N1 AND OT>0 THEN POSITION N1,N5:?
;6;"","";HI;""
3017 IF A=N1 AND OT=ST THEN POSITION N5,N7:?
;6;"(JOYSTICK)":GOTO TJ
3018 IF A=N1 AND OT>0 THEN POSITION N5,N7:?
;6;"(PADDLE)":GOTO TJ
3020 POSITION N1,N5:?
;6;"","";OFF;""
3021 IF TR=ST THEN POSITION N5,N7:?
;6;"(JOYSTICK)":GOTO TJ
3022 POSITION N5,N7:?
;6;"(PADDLE)":GOTO TJ
4000 FOR L3=N7*(N1+L1) TO Z STEP -N2:FOR L4=N10+N5 TO
GT-N5 STEP N10:POKE SH+N8,RND(Z)*TF:SOUND Z,L3,N10,L4
4001 NEXT L4:NEXT L3:POKE SH+N8,Z:SOUND Z,Z,Z,Z:RETURN
5000 FILL=USR(ADR(FS$),LM,UM,FI):RETURN

```

Memory Expansion Boards for the ATARI* Computer

Completely compatible with ATARI
hardware and software
No modifications necessary
Fully assembled and tested
User installable - simply plug it in
One year warranty
Gold connector tabs and sockets

16K Memory Board — \$ 54.95
32K Memory Board — \$124.95

Build Your Own Memory

16K board - no components - \$12.50
32K board — no components — \$30.00

Add \$2 Shipping and handling per board

*ATARI is a trademark of Atari Inc.

Dealer Inquiries Welcome

Tiny Tek, Inc.

P.O. Box 820249 • Dallas, TX 75382-0249
214-373-8926



HIGH SCORES

CROSSFIRE

Phil Heavin.....249,950

JAWBREAKER

Joe Caldwell....40,440

KEEP THOSE SCORES COMING IN!

UNCLASSIFIED ADS

ROOM FOR RAM

Trade-Atari Equipment; printer, disk drive, etc. in exchange for rent of large 1 bedroom apt. 28 mile and Van Dyke; call Jack Fraylick 781-9804

FOR SALE

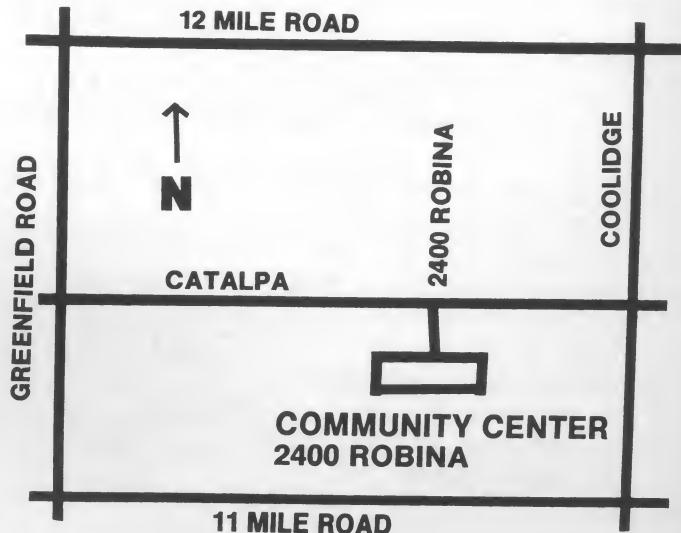
Atari Word Proc...\$89
2 16K RAM...49 ea.
Eastern Front (disk)...19
Datasoft Mail List (disk)...19
Comp. Card File (APX)...9
(or best offer)
Call Ken Vandellen, 821-5706

1982 MEETING SCHEDULE

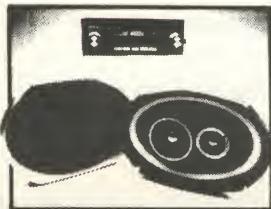
JULY 15
AUGUST 19
SEPTEMBER 16

OCTOBER 21
NOVEMBER 18
DECEMBER 16

BERKLEY COMMUNITY CENTER
2400 ROBINA
BERKLEY, MICHIGAN



Listen to what your money can buy at Tech Hifi.



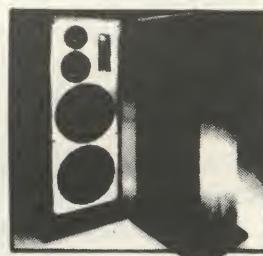
PIONEER
CAR STEREO

Pioneer car stereo to fit most cars is on sale, including the KP-5500 AM/FM stereo cassette player with Supertuner and the deluxe tri-axial TS-695 6x9" speakers. You can also drive a bargain on Sony, Jensen, Clarion, Mitsubishi, and Concord car stereos!



AIWA
PORTABLE STEREO

Top-quality Aiwa portable stereo is on sale, including the Aiwa CS-440 AM/FM stereo cassette recorder with 2 shortwave bands and world-wide voltage. And JVC, Hitachi, Panasonic, and Crown of Japan portables are priced to travel, too!

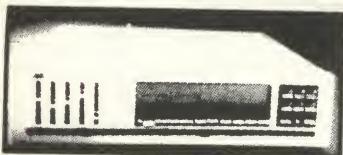
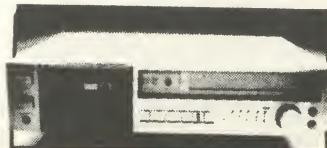


JBL SPEAKERS

Save on JBL speakers, like the magnificent R-133 3-way tower. Breakthrough speakers from Ohm, Celestion, EPI, Infinity, Cerwin-Vega, and TDC are clearance-priced too!

JBL

SONY CASSETTE DECKS



JVC RECEIVERS

the hifi bargain center

Tech Hifi will not be undersold!

We will meet or beat the price of any authorized dealers on any component they have in stock.

ASK ABOUT M.A.C.E. DISCOUNTS

28007 West Eight Mile Rd., Farmington 477-8810

Better price. Better advice.



BATS

By Stan Ockers

'Bats' is a program with a number of specialized techniques for the Atari. Perhaps you will find some of these useful in your programs.

The bats, stalactites and stalagmites are printed using special characters. These characters are created by reworking the normal character set (lines 100-104 and 32000-32108). The character set in ROM is first moved into RAM where it can be modified. Space is set aside by lowering RAMTOP five pages (line 32000), then going back up one page to calculate START. START is the beginning of the new character set of four pages (1K). The reason for leaving the blank page is that the display list and display data (residing at the top of memory) will in certain situations take in memory beyond RAMTOP.

Transferring 1K by peks and pokes is slow (it takes about 20 seconds), so I've written a machine language routine to do the job (listing 1). This routine is poked into a string ZZ\$ in line 100 and is called in line 32020 with the U S R function as : A = U S R (ADR(ZZ\$),ADDR1,ADDR2) where ADDR1 and ADDR2 are the 'from' and 'to' locations of the 1K block. Eight characters are changed including #%%&'()++ (lines 32030-32108).

For the player missile technique I refer you to Sheldon Leemon's excellent article referenced in line 139. The idea is to make up the player missile area using strings. String operations can then be used to erase players or move them vertically. The entire player missile area is cleared in line 170. The string P\$ falls in the player zero area (bat) while the string M\$ holds the missiles (insects). Images for the bat with wings down (BATDN\$) and up (BATUP\$) are also stored in strings. The image strings are positioned in the area strings at the proper spot for the current vertical position (lines 310-320). Bat images are moved one character at a time and blanks at each end insure that the previous image will be erased by the new.

The four insects are defined in string MM\$

(line 150) while their vertical positions are defined in line 287. The positions chosen are somewhat of a compromise, trying to keep the insects between stalactites and stalagmites as much as possible.

There are two types of stalactite - stalagmite pairs as defined in line 260. Line 270 provides copies of these in two other colors. If drawn as defined, the tips would be touching each other. Instead pairs are drawn in a loop (lines 500-570) by drawing part of the stalactite (to the tip), blanks equal to the width of the cave W and part of the stalagmite (from the tip).

The structure of the main part of the program is as follows:

Main Routine

Decrease width of cave (280)
 ; Next player (284-286)
 ; Position insects (287)
 ; Wait for signal to start bat (290-295)
 ; Print scorecard (300-301);
 ; Set bat at left side, clear coll. register
 (306)

For increasing horiz. positions:

; If trigger pressed, bat up one (310)
 ; If not pressed, down one (320)
 ; If hit poisioned insect go lose a bat
 (330)
 ; If another insect add 25 pts. (340-360)
 ; If hit stal. then lose a bat (370)
 ; Next horiz. pos.

Pass thru cave until round score>300
 (395);
 ; Add round score to total (400);
 ; Add bonus if earned (extra bat) (402)

Blink total (403)
 Until all players have gone thru at width W
 (404)

Lose a Bat Branch:

; Sound crash (420)
 ; Bat falls till another coll. (430)
 ; Remove bat,if last play dirge (445)
 ; If more players return to flashing score
 (460)
 ; Print game over etc. (470-472)

; Wait for start again (475)
; Reinitialize, start again (476)

You will notice that in none of the structure is there any mention of the horizontal insect movement. That's because it's automatic! Every 1/60th of a second Atari goes through an updating process. This occurs during the time the electron beam forming the TV picture is traveling from the bottom to the top of the screen, the so-called vertical blank period. Provision has been made to insert our own updating routine to take care of the missiles. The routine and the necessary code to insert it is given in listing 2. Both are poked into page six in lines 180 to 190. Four bytes are set aside in page six to hold the horizontal positions of the missiles. Every 1/60th of a second numbers are taken from these locations and put in the missile horizontal registers. By doing this during the time when the screen is blanked, the screen doesn't flicker.

This program handles a problem which has bothered me for some time. If you once

initialize PM graphics and then do a Graphics command (GR. 2), you end up with the player missles being vertical stripes which run the full length of the screen. Reversing the PM initialization doesn't seem to help. The problem is the chip which feeds the screen. It has some registers which are normally being filled by ANTIC during PM graphics. This chip can be disconnected from ANTIC (0 in 53277) but the registers still have information in them which result in vertical lines. These registers (53261-53264) have to be cleared with pokes as in line 1000. PM graphics is then reinitialized after the graphics mode change (line 2000).

A simple alteration of the display list improves the appearance of the display considerably. In this case, the four graphics 0 lines in the text area are changed to four graphics 1 lines (lines 2010-2040). You must remember that the lines are now 20 characters long instead of 40 and must alter your calculation of row and column positions in the text area accordingly. Notice in lines 290,300,301 and 700-800 that row 0 takes in the first two rows and row 1 the last two.

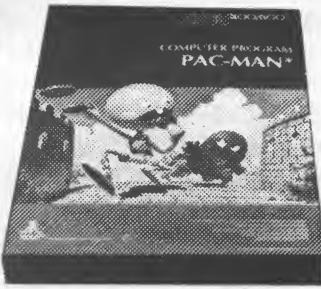
MegaRam

Hours: 9:30-8:30 M-Sat
12:00-5:00 Sun

COMPUTER SYSTEMS PAC-MAN COMES HOME AT LAST.



...and only \$32.95!
Our free catalog has over
400 programs for the Atari
at great
...DISCOUNTS...



--Write or call--

51280 VAN DYKE
NEAR 23 MILE RD
UTICA, MI (313) 254-2590

OTHER DEALS---

Intec 32K --- \$79.00

Printers...

C Itoh Comet I ... \$269.00

Prowriter ... \$489.00

! NAME	!
! ADDR	!
! CITY	STATE
! ZIP	!
! [] Send free catalog	
! [] Send Merchandise below	
!	!
!	!

```

20 REM *****
30 REM ** BATS **
40 REM ** S.O. 3-82 **
50 REM *****
100 DIM ZZ$(32):FOR I=1 TO 32:READ
A:ZZ$(I)=CHR$(A):NEXT I:GOSUB 32000:CLR
102 DATA 104,104,133,204,104,133,203,
104,133,206,104,133,205,162,4,160,0
104 DATA 177,203,145,205,136,208,
249,230,204,230,206,202,208,240,96
110 GOSUB 3000
130 TRAP 130?: "PLAYERS ";:POKE 764,255:INPUT NP
138 REM ** PM GRAPHICS **
139 REM ** SEE MAR '82 SOFTSIDE P. 69 **
140 DIM D$(1),F$((INT(ADR(D$)/1024)+1)
#1024-ADR(D$)-1),PM$(384),M$(128),P$(128),MM$(8)
150 RESTORE 160:FOR I=1 TO 8:READ
A:MM$(I)=CHR$(A):NEXT I
160 DATA 3,3,12,12,48,48,192,192
170 PM#=CHR$(0):PM$(384)=
CHR$(0):PM$(2)=PM$:M#=PM$:P#=M$
172 REM ** MISSILE COLORS **
174 POKE 704,14:POKE 705,39:POKE 706,54:POKE 707,70
179 REM ** VBI ROUT. TO MOVE MISSILES **
180 FOR I=1536 TO 1566:READ A:POKE I,A:NEXT I
190 DATA 104,160,14,162,6,169,7,76,
92,228,90,120,150,180,162,3,222,10,
6,189,10,6,157,4,208,202,16,244,
76,98,228
199 REM ** BAT IMAGES **
200 DIM BATDN$(5):BATDN$=P$:FOR I=2 TO 4:READ
A:BATDN$(I,I)=CHR$(A):NEXT I
210 DATA 24,165,66
220 DIM BATUP$(5):BATUP$=P$:FOR I=2 TO 4:READ
A:BATUP$(I,I)=CHR$(A):NEXT I
230 DATA 66,165,24
250 POKE 54279,ADR(PM$)/256:POKE 559,46:POKE
53277,3:POKE 623,4:A=USR(1536)
259 REM ** STALACTITES AND STALAGMITES **
260 DIM C$(42),U$(42):C$="*****"
*****&" :U$="") (*****'*****":FOR I=1 TO
14:C$(I+14)=CHR$(ASC(C$(I))-32)
270 U$(I+14)=CHR$(ASC(U$(I))-32)
:C$(I+28)=CHR$(ASC(C$(I))
+128):U$(I+28)=CHR$(ASC(U$(I))
+128):NEXT I
272 DIM P(NP),SCORE(NP),TOTAL(NP),
BN(NP),BONUS(NP)
275 W=7:P=0:POKE 82,0
278 FOR I=1 TO NP:SCORE(I)=0:TOTAL(I)=0
:BN(I)=3:BONUS(I)=1000:NEXT I:NXTCV=300*NP
279 REM ** CHANGE WIDTH OF CAVERN **
280 IF W>3 THEN W=W-1
282 GOSUB 480
284 P=P+1:IF P>NP THEN P=1
286 IF BN(P)=0 THEN 284
287 M#=PM$:FOR I=0 TO 3:M$(YST+5)*W+H
*(3-I)=MM$(2*I+1,2*I+2):NEXT I
289 REM ** MAIN LOOP **
290 POKE 656,1:POKE 657,22?: " Pull Joystick"
295 IF STICK(0)>13 THEN 295
299 REM ** SCORECARD **
300 ? CHR$(125):GOSUB 700:POKE 656,0:POKE 657,26?: "
PLAYER #";P
301 POKE 656,1:POKE 657,24?: "Round Total";:GOSUB 860
306 POKE 53248,30:YPOS=YST+20:POKE 53278,0:T=0:DIS=12
310 FOR XPOS=47 TO 200:POKE 53248,XPOS:IF STRIG(0)=0
THEN YPOS=YPOS-1:P$(YPOS)=BATUP$
320 IF STRIG(0)=1 THEN YPOS=YPOS+1:P$(YPOS)=BATDN$
330 IF PEEK(53256)>0 THEN POKE 1546,0:TOTAL(P)=TOTAL(P)-100:GOTO 420
340 IF PEEK(53257)>0 THEN POKE 1547,0:GOSUB 900
350 IF PEEK(53258)>0 THEN POKE 1548,0:GOSUB 900
360 IF PEEK(53259)>0 THEN POKE 1549,0:GOSUB 900
370 IF PEEK(53252)>0 THEN 420
390 NEXT XPOS:P#=PM$
395 IF SCORE(P)<300 THEN 306
400 TOTAL(P)=TOTAL(P)+SCORE(P)
:SCORE(P)=0:GOSUB 870
402 IF TOTAL(P)>BONUS(P) AND BN(P)<4 THEN
BONUS(P)=BONUS(P)+1
1000:BN(P)=BN(P)+1:GOSUB 700:DIS=10:T=30:GOSUB 630
403 FOR I=1 TO 30:GOSUB 880:FOR J=1 TO 30:NEXT J:GOSUB
860:NEXT I
404 IF P=NP THEN 280
410 GOTO 284
419 REM ** LOSE A BAT **
420 DIS=10:T=9:GOSUB 630
430 YPOS=YPOS+1:P$(YPOS)=BATDN$:POKE 53278,0:SOUND
1,YPOS,10,10:IF PEEK(53252)=0 THEN 430
445 GOSUB 640:P#=PM$:SCORE(P)=0
:BN(P)=BN(P)-1:GOSUB 700:IF BN(P)=0 THEN POKE
656,0:POKE 657,6? " ":"GOSUB 660
460 GOSUB 850:FOR I=1 TO NP:IF BN(I)>0 THEN 403
470 NEXT I:GOSUB 1000:GRAPHICS 17:POSITION 5,2:?
#6;"GamE OveR":FOR I=1 TO NP:POSITION 3,2+2*I:?
#6;"Player #";I;
472 ? #6;" = ";TOTAL(I):NEXT I:POSITION 3,23:?
#6;"PRESS ANY KEY";
475 FOR I=1 TO 300:NEXT I:GOSUB 655:IF FL=0 THEN 475
476 GOTO 275
479 REM ** DRAW CAVERN **
480 GOSUB 1000:GRAPHICS 2:GOSUB 2000:POKE 77,0
490 DL=INT(RND(0)*(8-W))+1:YST=8*(DL+1)
500 FOR X=0 TO 19:GOSUB 590:Y=0:FOR I=R+7-DL TO
R+6:POSITION X,Y?: #6;C$(I,I);Y=Y+1:NEXT I
510 FOR I=1 TO W:POSITION X,Y?: #6;" ";Y=Y+1:NEXT I
520 IF DL+W>10 THEN Y=Y-1:POSITION X,Y?: #6;" ";GOTO
540
530 GOSUB 590:FOR I=R TO R+9-DL-W:POSITION X,Y?: #
6;U$(I,I);Y=Y+1:NEXT I
540 IF DL<1 THEN DL=2:GOTO 570
550 IF DL>10-W THEN DL=9-W:GOTO 570
560 DL=DL+INT(RND(0)*3)-1
570 NEXT X
580 RETURN

```

continued

```

590 R=INT(RND(0)*6)*7+1:RETURN
629 REM ** SOUND SUBR'S **
630 FOR I=15 TO 0 STEP -1:SOUND 0,I,DIS,I:FOR J=1 TO
T:NEXT J:NEXT I:RETURN
640 FOR I=10 TO 2 STEP -2:SOUND 0,RND(0)*255,8,I:SOUND
1,RND(0)*255,8,I:FOR J=1 TO 30:NEXT J:NEXT I
650 SOUND 0,0,0,0:SOUND 1,0,0,0:RETURN
655 RESTORE 697:LS=30:LL=5:GOSUB 664:RETURN
660 RESTORE 690:LS=20:LL=10
664 FL=0
665 READ I,J:IF I=3 THEN RETURN
666 IF I=0 THEN 670
667 IF PEEK(53775)<255 THEN FL=1:RETURN
669 SOUND 0,I,10,10:SOUND 1,I-2,10,6
670 FOR I=1 TO J:FOR K=1 TO LS:NEXT K:NEXT I:SOUND
0,0,0,0:SOUND 1,0,0,0
675 FOR I=1 TO LL:NEXT I:GOTO 665
680 RESTORE 692:LS=12:LL=12:GOSUB 664
681 IF FL=1 THEN RETURN
682 RESTORE 694:GOSUB 664
683 IF FL=1 THEN RETURN
684 RESTORE 692:GOSUB 664
685 IF FL=1 THEN RETURN
686 RESTORE 696:GOSUB 664
687 IF FL=1 THEN RETURN
688 FOR I=1 TO 300:NEXT I:GOTO 680
690 DATA 243,4,243,4,243,1,243,4,
204,4,217,1,217,4,243,1,243,4,255,1,243,6,3,3
692 DATA 243,1,217,1,204,1,182,1,162,
1,204,1,162,1,0,1,173,1,217,1,173,1,
0,1,182,1,230,1,182,1,0,1
693 DATA 243,1,217,1,204,1,182,1,162,
1,204,1,162,1,121,1,3,3
694 DATA 136,1,162,1,204,1,162,1,136,
4,3,3
696 DATA 162,1,204,1,162,1,121,1,243,
4,3,3
697 DATA 81,4,85,2,102,1,108,1,121,6,
108,1,102,1,81,2,81,2,85,2,102,1,108,
1,121,8
698 DATA 108,2,91,2,102,2,108,2,121,1,
128,1,121,1,108,1,102,2,121,2,81,4,102,4,121,8,3,3
699 REM ** SUBR. TO INDICATE BATS LEFT **
700 POKE 656,0:POKE 657,6:?"":POKE 657,6:FOR
I=1 TO BN(P):? "+ ";:NEXT I:RETURN
850 POKE 656,1:POKE 657,5:? SCORE(P);"?":RETURN
860 POKE 656,1:POKE 657,12:? TOTAL(P);"?":RETURN
870 POKE 656,1:POKE 657,5:? "?":RETURN
880 POKE 656,1:POKE 657,12:? "?":RETURN
900 GOSUB 630:POKE 53278,0:SCORE(P)=SCORE(P)+25:GOTO
850
999 REM ** SUBR. TO REMOVE PM GR. **
1000 POKE 53277,0:POKE 54272,0:FOR I=53261 TO
53264:POKE I,0:NEXT I:RETURN
1999 REM ** SUBR. TO INSERT PM GR. **
2000 POKE 53277,3:POKE 559,46:START=(PEEK(106)+1):POKE
756,START

```

```

2009 REM ** ALTER DISPLAY LIST **
2010 A=PEEK(560)+256*PEEK(561)
2020 IF PEEK(A)>66 THEN A=A+1:GOTO 2020
2030 POKE A,70:POKE A+3,6:POKE A+4,6:POKE A+5,6
2040 RETURN
3000 ? ?:?:CHR$(125)? "+ + + + + + BATS + + + +
+ + "+?
3010 ? " The object is to fly your bat":? "through a
cavern eating flying"
3020 ? "insects along the way. The only":? "control
you have is the FIRE":? "button. Push it to go up."
3030 ? "Release it and your bat starts":? "to fall.":?
?:? " A round lasts 300 points or"
3035 ? "until you lose a bat. Each time":? "you hit a
stalactite or a"
3040 ? "stalagmite you lose one bat":? "and points
earned for that round."
3050 ? "Each insect you eat scores points":? "but do
not eat the bright insect the"
3060 ? "color of your bat.. it is poison!":? "Eat him
and you lose round points":? "plus 100 more and a
bat!":?
3065 ? "PRESS ANY KEY":GOSUB 680?:CHR$(125)? :?
3070 ? " After a time the cavern will":? "become
narrower and scores for the"
3080 ? "last round will be added to totals.":? "Every
1000 points you are given"
3090 ? "another bat, 4 maximum. Lose":? "all your
bats and you are out."
4000 ? "The game is over when everyone is":? "out of
bats. Everyone uses the":? "leftmost joystick":?
:RETURN
31999 REM ** CHANGE CHARACTER SET **
32000 POKE 106,PEEK(106)-5:GRAPHICS
0:START=(PEEK(106)+1)*256:POKE 756,START/256:POKE 752,1
32010 ? "INITIALIZING .....,"
32020 A=USR(ADR(ZZ$),57344,START)
:RESTORE 32100
32030 READ X:IF X=-1 THEN RESTORE :RETURN
32040 FOR Y=0 TO 7:READ Z:POKE X+Y+START,Z:NEXT Y:GOTO
32030
32100 DATA 32,255,255,127,127,126,62,
62,60
32101 DATA 40,60,28,28,24,8,8,8,8
32102 DATA 48,255,127,126,60,56,24,8,8
32103 DATA 56,8,24,28,124,124,254,254,
255
32104 DATA 64,60,126,126,126,126,126,
127,255
32105 DATA 72,16,16,16,16,16,24,60,60
32106 DATA 80,255,255,255,255,255,255,
255,255
32107 DATA 88,0,24,24,165,165,66,66,0
32108 DATA -1

```

CONTEST!

PUZZLE CORNER

By Charles Godfrey

This will be a new feature of our newsletter. Each month a new puzzle will be presented, with a review of the previous months' puzzle.

Problems will be selected that should require the writing of a computer program to find the answer. They are not meant to be trick questions, but practical problems with legitimate answers. To further entice you, M.A.C.E. Will award the first winner of each contest a MACE Library disk or tape of their choice!

What I would like (besides the right answer) is a few tidbits about how you solved the problem, such as: what computer did you use, what language, how many statements, and how long did it take the computer to arrive at the answer. Your puzzle master has solved this months problem using three different languages on as many different computers.

The correct entry bearing the earliest postmark will be considered the winner. No entries accepted after midnight, August 13, 1982. All entries become the property of MACE. Winner will receive a certificate good for one library disk or tape of their choice. In the event of a tie, more than one winner may be selected. Respond only by mail. No phone calls accepted.

Address your entry to:

MACE PUZZLER
2639 Hempstead
Auburn Heights, Michigan 48057

PUZZLER #1

You are sitting in the middle of a circle of 1000 people with a gun with 1000 bullets. Starting with anyone as number one, you shoot that person. You then start shooting every

other live person continuously around the circle until everyone is dead.

The puzzler is: BY NUMBER, WHO IS THE LAST ONE TO DIE?

This question was the problem-of-the-month several years ago at Yale university. If the wording bothers you, pretend that the people are dominoes. ☺



MICHIGAN ATARI COMPUTER ENTHUSIASTS

President: Arlan Levitan
12709 Borgman
Huntington Woods, MI 48070
399-6963
Micronet: 70675,463
Source ID: TCT987

Vice Pres.: Jerry Aamodt
4148 Huhn
Rochester, MI 48063
574-1020

Treasurer: Judy Braun
39281 Eliot
Mt. Clemens, MI 48043

Secretary: Sheldon Leemon
14400 Elm
Oak Park, MI 48237
398-2608

Programs: Gary Luzier
22530 Gordon
St. Clair Shores, MI 48081
773-3446

Librarians: Sam Findley (Disk)
8171 Denwood, Apt. #1
Sterling Heights, MI 48077
939-3822
Rodney Graham (Tape)
12270 Deming
Sterling Heights, MI 48077
264-6355

Newsletter: Marshall Dubin
2639 Hempstead
Auburn Heights, MI 48057
338-3488

PROGRAM LIBRARY SUBMISSION FORM

I wish to submit the following program to the M.A.C.E. program library. I warrant that I have good title to this program, and that it does not infringe upon any copyright. Limited rights are transferred herewith for the use of this program within the membership of M.A.C.E. and its associated Atari Computer organizations.

Program Name _____ Submission Media: Tape ____ Disk ____ List ____
Description _____

Catagory: Business ____ Demonstration ____ Education ____ Game ____ Utility ____ Other ____

Minimum System: Memory ____ K BASIC ____ Assembler ____ PILOT ____ PASCAL ____

Disk ____ Tape ____ Interface ____ Modem ____ Printer ____ Light Pen ____

Joystick Controllers ____ Paddle Controllers ____ Keyboard Cont. ____

Name _____ Phone _____ Date _____

M.A.C.E.
P.O Box 2785
Southfield, MI 48037

\$15.00 fee for 12 months*

M.A.C.E. MEMBERSHIP APPLICATION

Name _____ Phone _____

Street _____

City _____ State _____ Zip Code _____

Company (if applicable) _____

System Description _____ Disk/Tape _____

Suggestions _____

I can help with . . . _____

- New
- Renewal

*Make checks payable to:
M.A.C.E.

Date _____	Coupon _____
Amount _____	Membership Card Number _____
<input type="checkbox"/> Cash	Expires _____
<input type="checkbox"/> Check Number	



RITE WAY ENTERPRISES

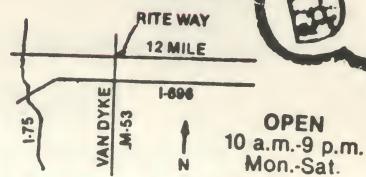
EVERYTHING DISCOUNTED EVERYDAY

ATARI
MICROSOFT
BASIC

\$ 75.95

(313) 751-2454
8262 12 MILE
WARREN, MI. 48093

THE
DISCOUNT
Computer
SOURCE



MICHIGAN ATARI COMPUTER ENTHUSIASTS
P.O. BOX 2785
SOUTHFIELD, MICHIGAN 48037

BULK RATE
U.S. POSTAGE
PAID
PERMIT #431
SOUTHFIELD, MI

IMPORTANT DATED MATERIAL

PLEASE DO NOT DELAY

Printing and Bindery Services by • GRAPHIC ENTERPRISES, INC. Detroit, Michigan • 313-839-6800